

# Multiscale modelling couples patches of nonlinear wave-like simulations

Meng Cao\*      A. J. Roberts†

April 28, 2014

## Abstract

The multiscale gap-tooth scheme is built from given microscale simulations of complicated physical processes to empower macroscale simulations. By coupling small patches of simulations over unsimulated physical gaps, large savings in computational time are possible. So far the gap-tooth scheme has been developed for dissipative systems, but wave systems are also of great interest. This article develops the gap-tooth scheme to the case of nonlinear microscale simulations of wave-like systems. Classic macroscale interpolation provides a generic coupling between patches that achieves arbitrarily high order consistency between the multiscale scheme and the underlying microscale dynamics. Eigen-analysis indicates that the resultant gap-tooth scheme empowers feasible computation of large scale simulations of wave-like dynamics with complicated underlying physics. As an pilot study, we implement numerical simulations of dam-breaking waves by the gap-tooth scheme. Comparison between a gap-tooth simulation, a microscale simulation over the whole domain, and some published experimental data on dam breaking, demonstrates that the gap-tooth scheme feasibly computes large scale wave-like dynamics with computational savings.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>The nonlinear microscale water wave model</b>	<b>3</b>
<b>3</b>	<b>Numerical simulations and eigenvalues show separation of scales</b>	<b>6</b>
<b>4</b>	<b>Coupling conditions connect patches across space</b>	<b>9</b>
4.1	Prove consistency for general wave systems . . . . .	9
4.2	Computer algebra establishes further consistency . . . . .	11
4.2.1	Algebraically confirm Theorem 1 . . . . .	11
4.2.2	Linear waves with dissipation . . . . .	12
4.2.3	Nonlinear wave systems . . . . .	13
4.3	Nonlinear slow manifolds exist . . . . .	14

---

\*School of Mathematical Sciences, University of Adelaide, South Australia 5005.  
<mailto:meng.cao@adelaide.edu.au> or <mailto:mengcao1188216@gmail.com>

†School of Mathematical Sciences, University of Adelaide, South Australia 5005.  
<mailto:anthony.roberts@adelaide.edu.au>

<b>5</b>	<b>Gap-tooth simulation of dam breaking</b>	<b>15</b>
5.1	Numerical gap-tooth simulation of dam breaking . . . . .	16
5.2	Boundary conditions for gap-tooth simulation . . . . .	19
<b>6</b>	<b>Conclusion</b>	<b>22</b>
<b>A</b>	<b>Ancillary computer algebra program</b>	<b>26</b>

# 1 Introduction

Mathematical equations describing floods and tsunamis are typically written at the macroscale of kilometres. But the level at which the underlying turbulent fluid physics are best understood is at the much finer sub-metre scale. Although macroscale models for floods and tsunamis are well established, for many multiscale and multiphysics wave-like problems good macroscale descriptions (good closures) do not exist. We aim to empower scientists and engineers to use brief bursts of microscale wave-like simulation on small patches of the space-time domain in order to make efficient accurate macroscale simulations without ever knowing a macroscale closure.

Many multiscale modelling techniques have been developed for dissipative systems (E & Engquist 2003, Kevrekidis et al. 2003, Roberts & Kevrekidis 2005, Hou et al. 2008, e.g.). Our macroscopic modelling further develops the equation-free gap-tooth scheme (Gear et al. 2003, Samaey, Kevrekidis & Roose 2005, Samaey, Roose & Kevrekidis 2005, Samaey et al. 2009, e.g.) to empower simulation of wave-like systems over large time and space scales from a *given* microscopic simulator. We suppose that the wave-like microscale simulator is computationally expensive so that only small time and spatial domain simulations are feasible: one example of future interest is direct numerical simulation of depth resolved turbulent fluid floods. The microscale simulator provides the necessary data for the macroscopic model, so whenever the microscale simulator improves, then the overall macroscale simulation will correspondingly improve.

This article develops and theoretically supports the gap-tooth method for general wave-like systems, and as an indicative application and test, applies the methodology to a Smagorinski model of turbulent shallow water flow. For example, Figure 1 exhibits the indicative gap-tooth simulation of a dam-break showing the water depth  $h(x, t)$ , where the microscale computations are only done in small patches of the spatial domain  $x$  over all time. Here the scheme uses microscale simulations of shallow water flow on small patches of space (Section 2), coupling the simulations over the intervening space, to simulate floods over a macroscale. Potential future applications could improve modelling of sediment erosion, transport, and deposition. The gap-tooth method and our theoretical support (Section 4) adapts to whatever microscale simulator is provided. The analysis of Section 4 indicates that the patches can occupy as small a fraction of space as is necessary for a good microscale simulation without affecting macroscale accuracy, thus indicating large computational gains are feasible with the methodology.

Previous research (Cao & Roberts 2013) explored classic linear non-dispersive waves and found classic interpolation on a macroscale staggered grid ensured ensure high order consistency to the linear wave equation. Section 4 proves for general linear microscale wave systems that the patch coupling condition (8) ensures arbitrarily high order consistency between the gap-tooth scheme and the underlying

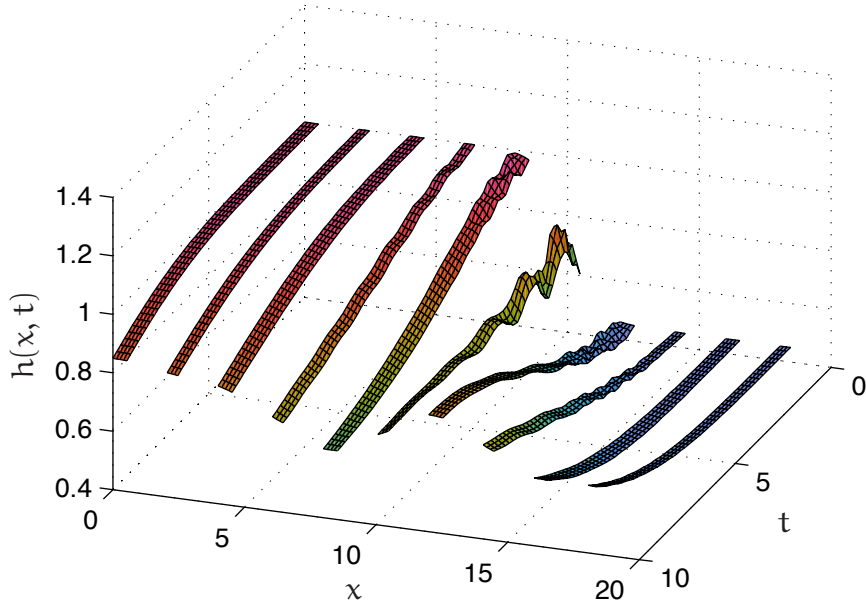


Figure 1: Indicative gap-tooth simulation of a dam-break shows the water depth  $h(x, t)$ : microscale computations are only done in small patches of the spatial domain  $x$ , but here over all time; at time zero (back) the dam breaks and at later times (front) the water forms a turbulent bore that propagates to the right.

microscale dynamics, and also establishes consistency for a class of nonlinear wave systems. Further, section 4.3 discusses nonlinear wave-like systems further and establishes the gap-tooth scheme with patches coupled with (8) have a slow manifold (Boyd 1995, Mackay 2004, e.g.) that forms the macroscale dynamics. This article focusses on waves in one spatial dimension with the expectation that generalisation to multiple space dimensions will be analogous to that for dissipative systems (Roberts et al. 2013, e.g.).

Section 3 implements the gap-tooth method by coupling small patches of the given microscale simulations of the shallow water Smagorinski model described in Section 2. Numerical eigenvalue analysis supports the theoretical results that there is an appropriate slow manifold of the macroscale dynamics in this application of the gap-tooth scheme.

Numerical simulations show that the gap-tooth coupling condition (8) works well for a range of shallow water flows. Section 5 applies the gap-tooth simulation to a dam-break (Figure 1), then compares the gap-tooth simulation with the microscale simulation over the whole domain, and with some experimental data of Stansby et al. (1998).

## 2 The nonlinear microscale water wave model

This section describes the nonlinear microscale simulator of the nonlinear shallow water wave PDE derived from the Smagorinski model of turbulent flow (Roberts 2008, Cao 2014). Often, wave-like systems are written in terms of two conjugate variables, for example, position and momentum density, electric and magnetic fields, and water depth  $h(x, t)$  and mean lateral velocity  $\bar{u}(x, t)$  as herein. This article uses the example of shallow water waves, but applies to any wave-like system

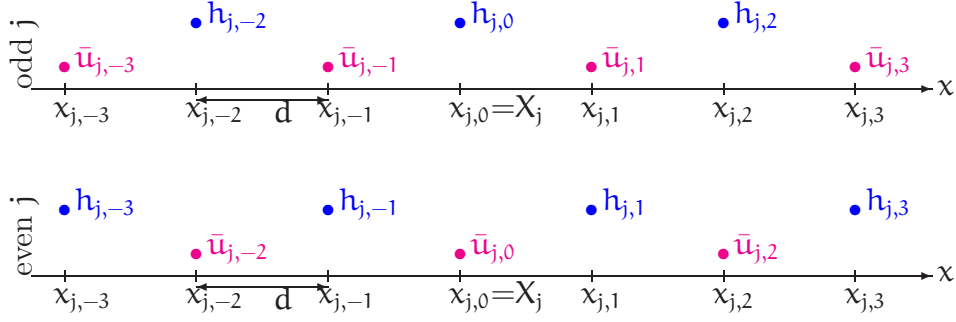


Figure 2: Scheme of the staggered grid points of the depth  $h_{j,i}$  (blue points) and velocity  $\bar{u}_{j,i}$  (magenta points) at the  $i$ th micro-grid point on the odd  $j$ th patch (top) and the even  $j$ th patch (bottom). This diagram shows the cases for  $n = 5$  interior grid points in each patch.

in the form

$$\frac{\partial h}{\partial t} = -c_1 \frac{\partial \bar{u}}{\partial x} + f_1[h, \bar{u}] \quad \text{and} \quad \frac{\partial \bar{u}}{\partial t} = -c_2 \frac{\partial h}{\partial x} + f_2[h, \bar{u}], \quad (1)$$

where the brackets indicate that the nonlinear functions  $f_\ell$  may involve various spatial derivatives of the fields  $h(x, t)$  and  $\bar{u}(x, t)$ . Specifically, this section invokes a nonlinear Smagorinski model of turbulent shallow water (Roberts 2008, Cao & Roberts 2012, e.g.) along an inclined flat bed: let  $x$  measure position along the bed and in terms of fluid depth  $h(x, t)$  and depth-averaged lateral velocity  $\bar{u}(x, t)$  the model PDEs are

$$\frac{\partial h}{\partial t} = -\frac{\partial(h\bar{u})}{\partial x}, \quad (2a)$$

$$\frac{\partial \bar{u}}{\partial t} = 0.985 \left( \tan \theta - \frac{\partial h}{\partial x} \right) - 0.003 \frac{\bar{u}|\bar{u}|}{h} - 1.045 \bar{u} \frac{\partial \bar{u}}{\partial x} + 0.26 h |\bar{u}| \frac{\partial^2 \bar{u}}{\partial x^2}, \quad (2b)$$

where  $\tan \theta$  is the slope of the bed. Equation (2a) represents conservation of the fluid. The momentum PDE (2b) represents the effects of turbulent bed drag  $\bar{u}|\bar{u}|/h$ , self-advection  $\bar{u}\partial\bar{u}/\partial x$ , nonlinear turbulent dispersion  $h|\bar{u}|\partial^2\bar{u}/\partial x^2$ , and gravitational hydrostatic forcing  $\tan \theta - \partial h/\partial x$ .

In practice, the microscale simulator will typically be either a spatial discretisation such as finite difference (Bijvelds et al. 1999, e.g.), finite element, or finite volume (LeVeque et al. 2011, e.g.), or a particle based method such as lattice Boltzmann (Liu et al. 2009, e.g.), molecular dynamics (Southern et al. 2008, e.g.), or smoothed-particle hydrodynamics (Monaghan 1992, e.g.). Our microscale simulator for the nonlinear PDEs (2) is a spatial discretisation on a fine-scale staggered grid within each patch. Figure 2 shows the staggered grid points of the depth  $h$  and depth-averaged velocity  $\bar{u}$  on a patch. Because we propose that the macroscale gap-tooth scheme employ a macroscale staggered grid, there are two types of alternating microscale patches corresponding to even and odd macroscale index  $j$ . In the  $j$ th patch, define a microscale staggered grid of spacing  $d$ : at the  $i$ th point of the micro-grid of the  $j$ th patch define the depth  $h_{j,i}$  when  $j - i$  is odd, and define the depth-averaged lateral velocity  $\bar{u}_{j,i}$  when  $j - i$  is even (Figure 2). Thus, approximate the Smagorinski shallow water PDEs (2) on the  $j$ th patch with centred

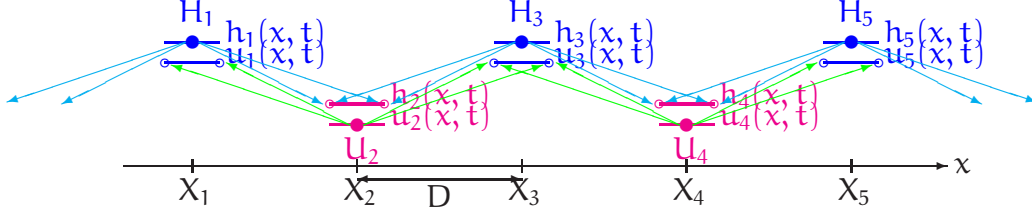


Figure 3: The macroscale scheme interpolates macroscale grid values  $H_j$  and  $U_j$  to provide edge values on each patch. The green arrows provide edge values of odd  $j$  patches by interpolating macroscale grid values  $U_j$ . The cyan arrows provide edge values of even  $j$  patches by interpolating macroscale grid values  $H_j$ .

differences in microscale space as the discrete

$$\frac{\partial h_{j,i}}{\partial t} = -\frac{(h_{j,i+2} + h_{j,i})\bar{u}_{j,i+1}}{4d} + \frac{(h_{j,i-2} + h_{j,i})\bar{u}_{j,i-1}}{4d}, \quad (3a)$$

$$\begin{aligned} \frac{\partial \bar{u}_{j,i}}{\partial t} = & 0.985 \left( \tan \theta - \frac{h_{j,i+1} - h_{j,i-1}}{2d} \right) - 0.003 \frac{\bar{u}_{j,i}|\bar{u}_{j,i}|}{h_{j,i}} \\ & - 1.045 \bar{u}_{j,i} \frac{\bar{u}_{j,i+2} - \bar{u}_{j,i-2}}{4d} + 0.26 h_{j,i} |\bar{u}_{j,i}| \frac{\bar{u}_{j,i+2} - 2\bar{u}_{j,i} + \bar{u}_{j,i-2}}{4d^2}, \end{aligned} \quad (3b)$$

where  $d$  is the microscale spatial step within a patch. Such a microscale lattice simulator is known to be consistent with the PDEs (2) to a error  $\mathcal{O}(d^2)$  which typically is negligible for small patches.

The dam break shown in Figure 1 was generated by such microscale simulations in patches coupled together across unresolved space. Time integration was done by Matlab `ode15s`.

The macroscale grid is also staggered: Figure 3 shows the alternating patches. Consequently, the macroscale model is to be parametrised by the macroscale grid values

$$U_j(t) := \bar{u}_j(X_j, t) \text{ for even } j, \text{ and } H_j(t) := h_j(X_j, t) \text{ for odd } j, \quad (4)$$

as shown in Figure 3. Following the pilot study of Cao & Roberts (2013), for the specific analysis and simulations of Sections 3 and 5 cubic interpolation of macroscale values from the nearest three patches on either side of a patch provided boundary values on that patch. That is, on patches with odd index  $j$  the boundary value of the microscale  $\bar{u}$ -field are determined as

$$\bar{u}_j(X_j \pm rD, t) = [\mu \pm \frac{1}{2}r\delta + \frac{1}{8}(-1 + r^2)\mu\delta^2 \pm \frac{1}{48}(-r + r^3)\delta^3] U_j(t), \quad (5a)$$

for centred mean and difference operators,  $\mu U_j := (U_{j+1} - U_{j-1})/2$  and  $\delta U_j := U_{j+1} - U_{j-1}$  respectively (the staggered macroscale grid of Figure 3 requires these non-standard definitions). Correspondingly, on patches with even index  $j$  the boundary value of the microscale  $h$ -field are

$$h_j(X_j \pm rD, t) = [\mu \pm \frac{1}{2}r\delta + \frac{1}{8}(-1 + r^2)\mu\delta^2 \pm \frac{1}{48}(-r + r^3)\delta^3] H_j(t). \quad (5b)$$

The expansion (6) justifies that these are cubic interpolation of the four surrounding macroscale grid values. This interpolation couples the macroscale staggered grid of patches across un-simulated space, as seen in the dam break of Figure 1, to form a well-posed simulation.

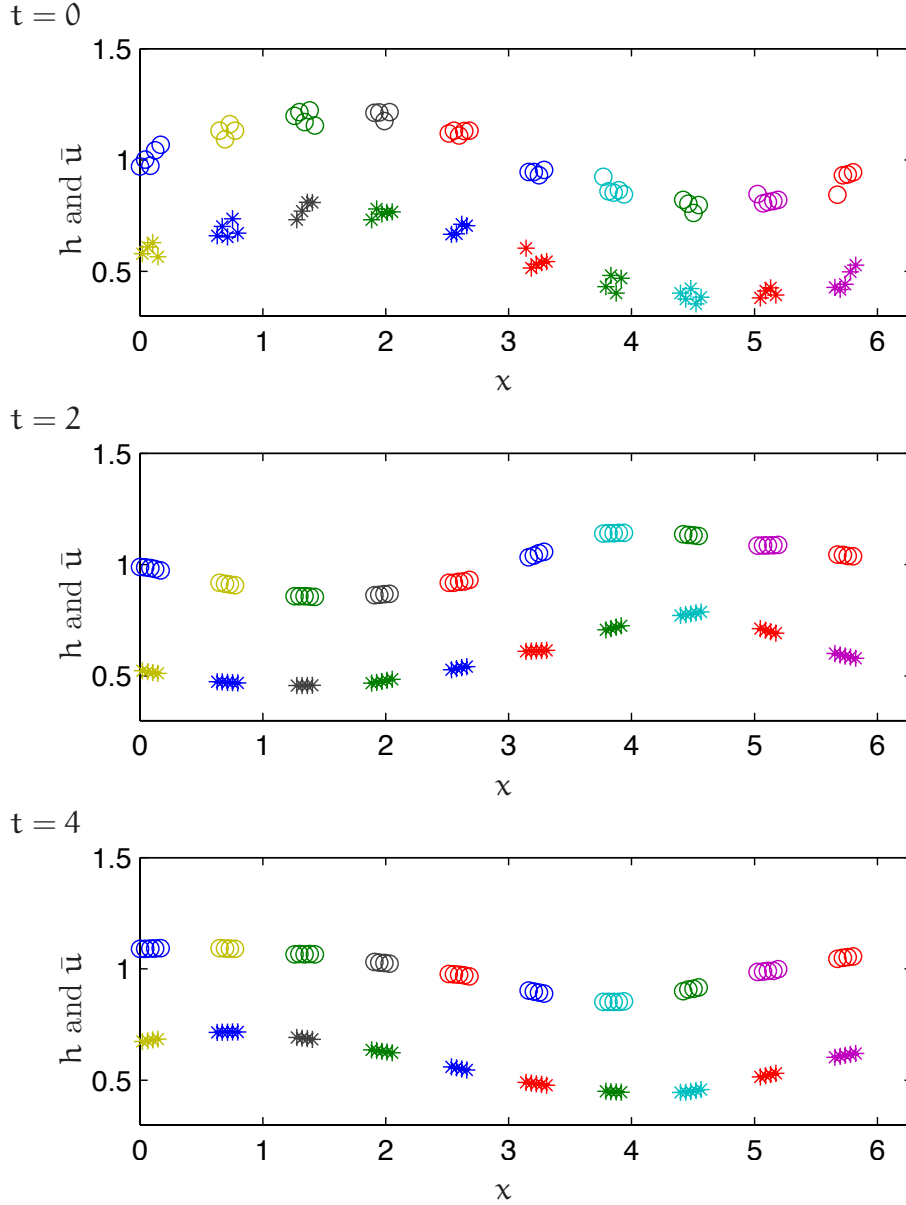


Figure 4: Gap-tooth simulation of water depth  $h$  (circles) and average lateral velocity  $\bar{u}$  (stars) by the nonlinear microscale simulator (3) and the coupling conditions (5) on domain  $\Omega = [0, 2\pi]$  via  $m = 10$  patches and  $n = 9$  microscale grid points on each patch at three times. The patch size ratio is  $r = 1/6$  and the mean slope of the bed is  $\tan \theta = 0.001$ .

### 3 Numerical simulations and eigenvalues show separation of scales

We numerically explore the macroscale turbulent fluid flow on a slightly inclined flat bed using the gap-tooth scheme with the nonlinear microscale simulator (3) and the cubic coupling conditions (5). Numerical simulations are straightforwardly implemented for equations (3) on staggered grids in space of Figures 2 and 3. Time integration was performed by Matlab's `ode15s`.

For example, we present the case of  $2\pi$ -periodic dynamics simulated by  $m = 10$  patches in a period, and  $n = 9$  interior microscale grid points on each patch. The patch size ratio was  $r = 1/6$  which means about a third of the spatial domain

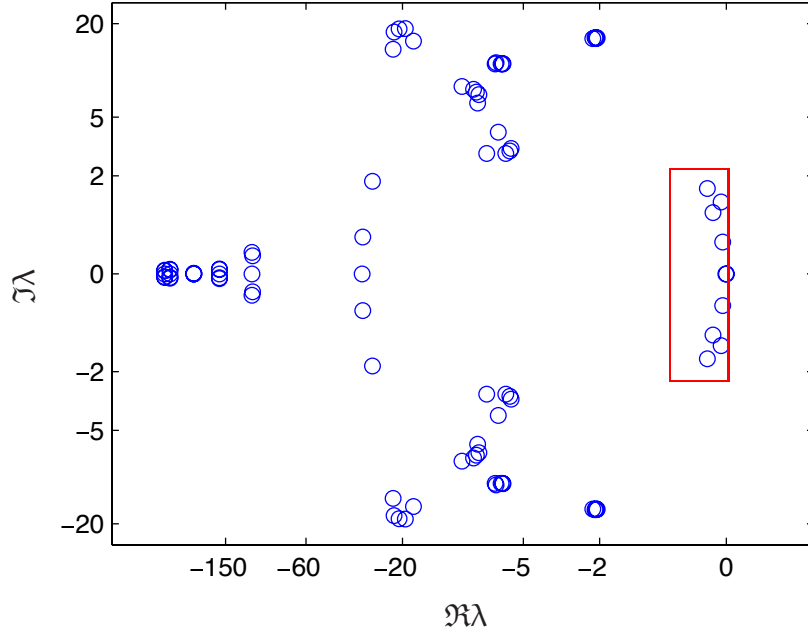


Figure 5: Distribution of the real and imaginary parts of the numerical eigenvalues (non-uniform scaling of axes) for flood PDEs (3) with  $m = 10$  patches and  $n = 9$  microscale grid points on each patch. The domain  $\Omega$  is assumed  $2\pi$ -periodic, the length scale ratio  $r = 1/6$ , and the bed slope  $\tan \theta = 0.001$ .

is covered by patches, and about two-thirds is unsimulated space: in practice we aim use smaller  $r$  but this  $r$  shows structures more clearly. Setting the bed slope to  $\tan \theta = 0.001$  an equilibrium flow of equation (2b) is that depth  $h = 1$  and depth-averaged lateral velocity  $\bar{u} \approx 18.1 \tan^{1/2} \theta = 0.57$  (non-dimensional). Figure 4 plots a numerical gap-tooth simulation for the depth  $h$  and depth-averaged lateral velocity  $\bar{u}$  at three times. At the initial time  $t = 0$ , we superimposed on the equilibrium flow a macroscale wave of  $0.2 \sin x$  together with small random microscale noise. The  $t = 2$  graph shows that the microscale structures within a patch has smoothed quickly by the microscale dissipation  $h|\bar{u}|\partial^2 \bar{u}/\partial x^2$ . In addition, the macroscale wave propagates downstream on the free surface, decaying slowly, as illustrated by the  $t = 4$  graph.

Eigenvalue analysis illuminates the gap-tooth scheme when applied to non-linear water wave microscale discretisation (3) with the cubic coupling conditions (5). Let's consider further the waves upon the equilibria of water flowing down an inclined plane with bed drag balancing gravitational forcing. By the non-dimensionalisation we just consider the equilibrium of water depth  $h_{j,i} = 1$  and depth-averaged lateral velocity  $\bar{u}_{j,i} \approx 18.1 \tan^{1/2} \theta$ . The spectrum of the Jacobian of the system characterises the dynamics in the neighbourhood of this equilibrium. We estimated the Jacobian to about seven digits accuracy via centred numerical differentiation of the gap-tooth simulation procedure. Then standard routines computed the complete spectrum of eigenvalues.

For example, to match the simulation of Figure 4 we analysed the case of  $2\pi$ -periodic dynamics predicted by  $m = 10$  patches in a period, and  $n = 9$  interior microscale grid points on each patch to give a system in 90 variables. The patch size ratio was  $r = 1/6$ . Figure 5 plots the growth rate ( $\Re\lambda$ ) and frequency ( $\Im\lambda$ ) obtained from the Jacobian. There are 40 pairs of complex conjugate eigenvalues with large negative real parts which represent the microscale modes within the patches. Most of these negative real parts are between  $-2$  and  $-200$ . These



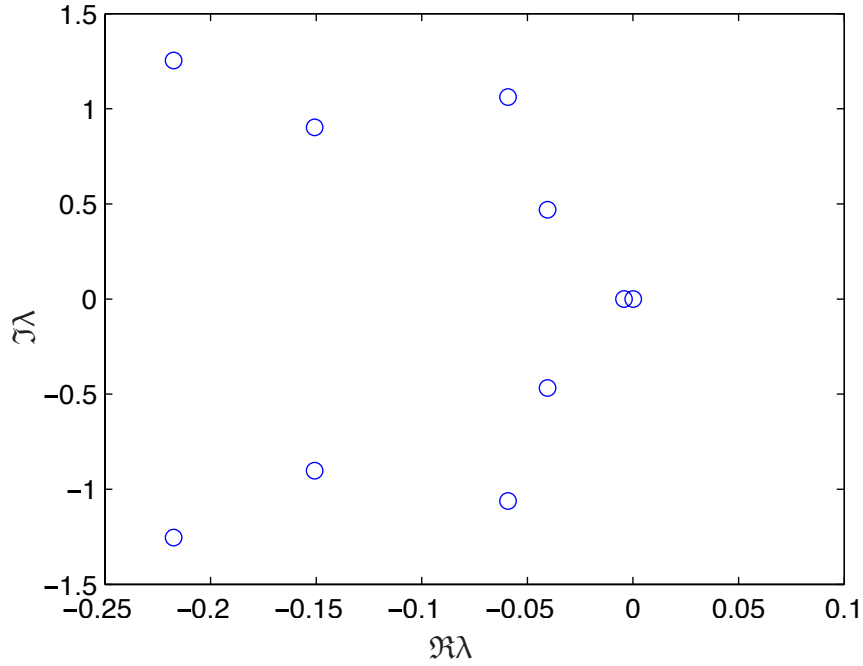


Figure 6: Zoom into the values in the red box in Figure 5. There are four complex conjugate pairs of values, and another two real values 0.0002 and  $-0.0042$ .

represent microscale waves within the patches which decay rapidly through the microscale turbulent dissipation, dominantly  $\mathbf{h}|\bar{\mathbf{u}}|\partial^2\bar{\mathbf{u}}/\partial\mathbf{x}^2$ . The large imaginary parts of many of these eigenvalues reflect the fast oscillation of the microscale waves within the patches.

With  $\mathbf{m} = 10$  patches on a staggered macroscale grid the scheme resolves four macroscale waves: two propagating upstream and two downstream. The eigenvalues with small real parts in the red box of Figure 5 represent these macroscale waves. Figure 6 zooms in these small eigenvalues and shows that there are four pairs of small decay, the different imaginary-part (frequencies) distinguish upstream and downstream propagation, together with two real values near zero (0.0002 and  $-0.0042$ ). This slow decay of the four macroscale waves is due to a combination of the turbulent dissipation and the small nonlinear bed drag  $0.003\bar{\mathbf{u}}|\bar{\mathbf{u}}|/\mathbf{h}$ . The small negative eigenvalue of  $-0.0042$  represents the decay of uniform flow to the equilibrium speed down the sloping bed. The smallest eigenvalue is zero to numerical error in the Jacobian and represents conservation of water. This spectrum of eigenvalues confirms the gap-tooth scheme models the macroscale wave propagation without the scheme explicitly knowing any macroscale closure.

The gap between the growth rate  $\Re\lambda \approx 0$  and  $\Re\lambda \approx -2$ , as shown in Figure 5, characterises the separation between slow macroscale dynamics and the fast microscale dynamics. As discussed in section 4.3, this gap indicates that there is a nonlinear slow manifold of the macroscale modes in the gap-tooth dynamics (Roberts 1988, Chicone 2006, Potzsche & Rasmussen 2006, e.g.). Sections 4.1–4.2 first establish the high order consistency between the macroscale gap-tooth scheme and the underlying microscale system.



## 4 Coupling conditions connect patches across space

We propose and analyse a macroscale staggered grid of patches as shown in Figure 3. This section establishes that classic macroscale interpolation provides a coupling between patches that achieves arbitrarily high order consistency between the patch scheme and the underlying microscale dynamics for general microscale systems.

### 4.1 Prove consistency for general wave systems

For definiteness in theoretical support, let there be  $m$  patches in a spatial domain  $\Omega$  with the fields required to be periodic in  $x$ , and the fields to be in the Sobolev space  $\mathbb{H}^2(\Omega)$  of square-integrable functions on  $\Omega$  with square-integrable first two derivatives. We model the dynamics on patches  $E_j$  of an equi-spaced macroscale grid  $X_j = jD$ : define the patches  $E_j := \{x \in \Omega \mid |x - X_j| < rD\}$ , for  $j = 1, \dots, m$ , centred on each macroscale grid point, and the collection of patches  $E := \{E_j \mid j = 1, \dots, m\}$ . The parameter  $r$  is the ratio between a patch half-width and the macroscale step  $D$ . The parameter  $r$  characterises the size of each patch relative to the distance between neighbouring patches: when  $r = 1/2$  the neighbouring patches meet; and when  $r = 1$  the patches overlap as was found so useful in holistic discretisation Roberts (2001, 2003). When the ratio  $r$  is small, the patches form a relatively small part of the physical domain to engender a computationally efficient scheme for multiscale simulation.

To couple patches of microscale simulations, define a macroscale shift operator  $\mathcal{E}h(x) := h(x+2D)$ <sup>1</sup> and corresponding centred difference and mean operators here defined as  $\delta := \mathcal{E}^{1/2} - \mathcal{E}^{-1/2}$  and  $\mu := (\mathcal{E}^{1/2} + \mathcal{E}^{-1/2})/2$ . Consider a field  $h(x) \in \mathbb{H}^2(\Omega)$ . Evaluating the field  $h$  at a shift corresponding to the width of a patch gives

$$\begin{aligned} h(x \pm rD) &= \mathcal{E}^{\pm r/2} h \\ &= (1 \pm \mu\delta + \tfrac{1}{2}\delta^2)^{r/2} h \quad (\text{Natl Physical Lab 1961, p.65, e.g.}) \\ &= \frac{\mu}{\sqrt{1 + \delta^2/4}} (1 \pm \mu\delta + \tfrac{1}{2}\delta^2)^{r/2} h \quad (\text{as } \mu^2 = 1 + \delta^2/4) \\ &= [\mu \pm \tfrac{1}{2}r\delta + \tfrac{1}{8}(-1 + r^2)\mu\delta^2 \pm \tfrac{1}{48}(-r + r^3)\delta^3 + \dots] h, \end{aligned} \quad (6)$$

This expansion motivates defining the corresponding ameliorated coupling operators

$$\begin{aligned} \mathcal{C}_{\pm} &:= \gamma [\mu \pm \tfrac{1}{2}r\delta] + \gamma^3 [\tfrac{1}{8}(-1 + r^2)\mu\delta^2 \pm \tfrac{1}{48}(-r + r^3)\delta^3] \\ &\quad + \gamma^5 [\tfrac{1}{384}(9 - 10r^2 + r^4)\mu\delta^4 \pm \tfrac{1}{3840}(9r - 10r^3 + r^5)\delta^5] \\ &\quad + \gamma^7 [\tfrac{1}{46080}(-225 + 259r^2 - 35r^4 + r^6)\mu\delta^6 \\ &\quad \pm \tfrac{1}{645120}(-225r + 259r^3 - 35r^5 + r^7)\delta^7] + \dots, \end{aligned} \quad (7)$$

that operates on the patch index  $j$ , in terms of mean and difference operators that hereafter operate on index  $j$ : for example,  $\delta H_j = H_{j+1} - H_{j-1}$  and  $\mu H_j = (H_{j+1} + H_{j-1})/2$ . The parameter  $\gamma$  conveniently labels the spatial extent of the various terms appearing in coupling operators (7) (discussed in the next paragraph). Then, as suggested by Figure 3, we require boundary conditions for each patch of

$$\bar{u}_j(X_j \pm rD, t) = \mathcal{C}_{\pm} \bar{u}_j(X_j, t) \quad \text{and} \quad h_j(X_j \pm rD, t) = \mathcal{C}_{\pm} h_j(X_j, t), \quad (8)$$

<sup>1</sup>The full shift  $\mathcal{E}$  skips a patch because we employ a staggered macroscale grid as shown in Figure 3. Expressions for mean and difference operators have to be correspondingly adjusted for the staggered grid.

for odd and even  $j$  respectively. By the definition of macroscale values (4), the right-hand sides of the boundary conditions (8) couple the microscale patches together via interpolation of the macroscale grid values.

We obtain various accuracies for the macroscale simulation by truncating the coupling (8) to various orders in the label  $\gamma$ . This follows as  $\gamma$  parametrises the stencil width of the interpolation. For example, truncating to errors  $\mathcal{O}(\gamma^3)$  gives linear interpolation from the nearest neighbour patches,  $h_j(X_j \pm rD, t) = (\mu \pm \frac{1}{2}r\delta)H_j$ , as Figure 3 illustrates. Whereas truncating to errors  $\mathcal{O}(\gamma^5)$  gives cubic interpolation from nearest and next nearest patches. Truncating to errors  $\mathcal{O}(\gamma^7)$  gives a quintic interpolation, and so on. One key property of the coupling is that, when truncated to errors  $\mathcal{O}(\gamma^p)$ , and upon setting label  $\gamma = 1$ , the difference  $\mathcal{C}_+ - \mathcal{C}_- = \mathcal{E}^{r/2} - \mathcal{E}^{-r/2} + \mathcal{O}(\delta^{p-1})$  in the limit of large length scale macroscale variations. Next we prove that the various order interpolations of the gap-tooth scheme achieves corresponding orders of consistency with the microscale system over  $\Omega$ —a consistency analogous to that for dissipative systems (Roberts & Kevrekidis 2005, Roberts et al. 2011, e.g.).

**Theorem 1.** *Consider the general coupled system of two equations*

$$\partial_t h = \mathcal{L}_1 \bar{u} + c_1 h \quad \text{and} \quad \partial_t \bar{u} = \mathcal{L}_2 h + c_2 \bar{u}, \quad (9)$$

for fields  $h(x, t), \bar{u}(x, t) \in \mathbb{H}^2(\Omega)$ , for generic odd, homogeneous linear operators  $\mathcal{L}_1$  and  $\mathcal{L}_2$ , and constants  $c_1$  and  $c_2$ . Let  $h_j(x, t), \bar{u}_j(x, t) \in \mathbb{H}^2(E_j)$  denote the subgrid fields on the  $j$ th patch satisfying (9) in patches  $E_j$  with the coupling conditions (8). When inter-patch coupling (8) is truncated to residuals  $\mathcal{O}(\gamma^p)$ , then the macroscale grid values (4) evolve consistently with the equations (9) to errors  $\mathcal{O}(\delta^{p-1})$  (upon setting  $\gamma = 1$ ).

In this theorem the linear operators  $\mathcal{L}_\ell$  could represent partial derivatives, microscale discretisations, lattice Boltzmann interactions, and so on. The following proof is based upon the ideas used to prove analogous consistency for dissipative systems (Roberts et al. 2011, Theorem 7).

*Proof.* The proofs for each of the equations (9) are the same with appropriate interchange of symbols. Consider  $\partial_t \bar{u} = \mathcal{L}_2 h + c_2 \bar{u}$ . Because the linear operator  $\mathcal{L}_2$  is odd and homogeneous, we formally expand the operator

$$\mathcal{L}_2 = \sum_{k=0}^{\infty} \ell_{2k+1} \delta_r^{2k+1} = \ell(\delta_r), \quad (10)$$

in terms of the patch sized, microscale, centred difference  $\delta_r := \mathcal{E}^{r/2} - \mathcal{E}^{-r/2}$ , for some coefficients  $\ell_{2k+1}$  and corresponding function  $\ell$ . By the term ‘generic odd’ in the theorem, we mean the coefficient  $\ell_1 \neq 0$ . The second of (9) on the patch  $E_j$  determines  $\partial_t \bar{u}_j = \mathcal{L}_2 h_j + c_2 \bar{u}_j$  and now becomes  $(\partial_t - c_2) \bar{u}_j = \ell(\delta_r) h_j$ . Because  $\ell_1 \neq 0$ , function  $\ell$  has a smooth inverse function  $\ell^{-1}$ , at least near zero, and so we rearrange this microscale equation to

$$\ell^{-1}(\partial_t - c_2) \bar{u}_j = \delta_r h_j. \quad (11)$$

Now evaluate (11) at the patch centre  $x = X_j$ : on the left-hand side the time derivatives commute with the evaluation at  $x = X_j$  so equation (11) becomes, by definition (4),

$$\ell^{-1}(\partial_t - c_2) \bar{u}_j = h_j(X_j + rD, t) - h_j(X_j - rD, t)$$

$$\begin{aligned}
&= \mathcal{C}^+ h_j(X_j, t) - \mathcal{C}^- h_j(X_j, t) \quad (\text{by coupling (8)}) \\
&= (\mathcal{C}^+ - \mathcal{C}^-) H_j \quad (\text{by definition (4)}) \\
&= (\mathcal{E}^{+r/2} - \mathcal{E}^{-r/2}) H_j + \mathcal{O}(\delta^{p-1} H_j) \quad (\text{by truncating (7) with } \gamma = 1) \\
&= \delta_r H_j + \mathcal{O}(\delta^{p-1} H_j), \tag{12}
\end{aligned}$$

where  $p$  is the order of error in  $\gamma$  of the truncated coupling operators (7) (if order  $p$  is even then the error is  $\mathcal{O}(\delta^p H_j)$ ). Equation (12) is a closed relation among the macroscale quantities. Reverting the inverse function  $\ell^{-1}$ , and equation (12) implies

$$(\partial_t - c_2) U_j = \ell(\delta_r) H_j + \mathcal{O}(\delta^{p-1} H_j) = \mathcal{L}_2 H_j + \mathcal{O}(\delta^{p-1} H_j),$$

which then becomes

$$\partial_t U_j = \mathcal{L}_2 H_j + c_2 U_j + \mathcal{O}(\delta^{p-1} H_j). \tag{13}$$

Similarly for the companion equation of (9). That is, in the patch scheme with coupling conditions (8), the macroscale grid values (4) evolve consistently to any specified order with the microscale system (9) solved on the whole domain  $\Omega$ .  $\square$

## 4.2 Computer algebra establishes further consistency

The previous subsection established consistency for general linear wave systems with simple drag, whereas we generally want to apply the patch scheme to wave systems with other dissipative mechanisms, and to nonlinear systems. This section uses computer algebra to show that consistency is also obtained for a variety of such interesting systems.

### 4.2.1 Algebraically confirm Theorem 1

Consider the following dispersive system in the wave-like form (9):

$$\frac{\partial h}{\partial t} = c_1 h - \frac{\partial \bar{u}}{\partial x} - c_{11} \frac{\partial^3 \bar{u}}{\partial x^3} \quad \text{and} \quad \frac{\partial \bar{u}}{\partial t} = c_2 \bar{u} - \frac{\partial h}{\partial x} - c_{21} \frac{\partial^3 h}{\partial x^3}, \tag{14}$$

with constant coefficients  $c_1, c_{11}, c_2, c_{21}$ . We confirm the consistency, established by Theorem 1, between the gap-tooth scheme and this underlying microscale system. Computer algebra constructs solutions to the system (14) on the patches  $E_j$  when coupled by (8) as a regular power series in the coupling parameter  $\gamma$ . In the solution, terms of up to  $\gamma^{p-1}$  then encode all the effects of truncating the coupling condition (8) to errors  $\mathcal{O}(\gamma^p)$ ; that is, to account for interactions between a patch and its  $p$  neighbours on either side.

For example, the computer algebra of Appendix A (with `choice:=1`) derives the microscale field in each patch as  $(h_j, \bar{u}_j) = \mathcal{M}(H_j, U_j)$  where, in terms of  $\xi = (x - X_j)/(rD)$ , operator

$$\mathcal{M} = \begin{cases} 1 + \gamma^2 \left[ \frac{1}{2} r \xi \delta \mu + \frac{1}{8} r^2 \xi^2 \delta^2 \right] + \mathcal{O}(\gamma^4), \\ \gamma \left[ \mu + \frac{1}{2} r \xi \delta \right] + \gamma^3 \left[ \frac{1}{8} (-1 + r^2 \xi^2) \delta^2 \mu + \frac{1}{48} (-r \xi + r^3 \xi^3) \delta^3 \right] + \mathcal{O}(\gamma^4), \end{cases}$$

alternating upon whether  $j$  is even or odd and whether applied to  $h_j$  or  $\bar{u}_j$ . Remarkably, in the class of PDEs (14) the microscale field is independent of the coefficients  $c_k$ : this independence does not generally occur in the other classes of

PDEs. For these microscale fields, the computer algebra, Appendix A, derives the corresponding evolution of the macroscale values to be, for the appropriate  $k$ ,

$$(\dot{H}_j, \dot{U}_j) = \left[ -\frac{1}{D} \left( \frac{1}{2} \gamma \delta - \frac{1}{48} \gamma^3 \delta^3 \right) - \frac{1}{8D^3} c_{k1} \gamma^3 \delta^3 \right] (U_j, H_j) + c_k (H_j, U_j) + \mathcal{O}(\gamma^4), \quad (15)$$

These macroscale evolution equations correspond to a conventional macroscale discretisation of the microscale PDEs (14). But remember that the gap-tooth scheme would generate a macroscale simulation obeying (15) without knowing explicitly such a closure.

The required high order consistency to confirm Theorem 1 is explored by transforming such macroscale discrete models (15) to its equivalent PDE and comparing to the microscale PDE (14). Truncating the coupling conditions (8) to errors  $\mathcal{O}(\gamma^9)$ , in a couple of CPU seconds the computer algebra program of Appendix A derives the higher order version of the macroscale model (15). Post-processing then uses Taylor series,  $H_{j+p} = \sum_{n=0}^{\infty} (pD)^n / n! \partial^n H / \partial X^n$ , to transform the higher order version of (15) to the equivalent PDEs for the macroscale variables as

$$\begin{aligned} \frac{\partial H}{\partial t} = & - \left[ \gamma \frac{\partial U}{\partial X} + \frac{1}{6} (\gamma - \gamma^3) D^2 \frac{\partial^3 U}{\partial X^3} + \frac{1}{120} (\gamma - 10\gamma^3 + 9\gamma^5) D^4 \frac{\partial^5 U}{\partial X^5} \right. \\ & \left. + \frac{1}{5040} (\gamma - 91\gamma^3 + 315\gamma^5 - 225\gamma^7) D^6 \frac{\partial^7 U}{\partial X^7} \right] \\ & - c_{11} \left[ \gamma^3 \frac{\partial^3 U}{\partial X^3} + \frac{1}{2} (\gamma^3 - \gamma^5) D^2 \frac{\partial^5 U}{\partial X^5} + \frac{1}{120} (13\gamma^3 - 50\gamma^5 + 37\gamma^7) D^4 \frac{\partial^7 U}{\partial X^7} \right] \\ & + c_1 H + \mathcal{O}((D^8 + c_{11} D^6) \frac{\partial^9 U}{\partial X^9}), \end{aligned} \quad (16)$$

and similarly for  $\partial U / \partial t$ . Observe in these expressions how beautifully various contributions cancel when artificial parameter  $\gamma$  is set to one: the result is that the macroscale variables  $H$  and  $U$  evolve consistently with the microscale PDEs (14). However, when the coupling between patches is limited to  $p-1$  nearest neighbouring patches on either side, equivalent to truncating the coupling (7) to errors  $\mathcal{O}(\gamma^p)$ , then the PDE (16) confirms the consistency holds to errors  $\mathcal{O}(\partial^p U / \partial X^p)$  for  $p \in \{3, 5, 7, 9\}$ ; that is, to errors  $\mathcal{O}(\delta^p U_j)$  in accord with Theorem 1.

#### 4.2.2 Linear waves with dissipation

Consider the following coupled system supporting dispersive waves that are damped by a diffusion of strength  $c_3$  and  $c_4$ :

$$\frac{\partial h}{\partial t} = -\frac{\partial \bar{u}}{\partial x} - c_{11} \frac{\partial^3 \bar{u}}{\partial x^3} + c_3 \frac{\partial^2 h}{\partial x^2} \quad \text{and} \quad \frac{\partial \bar{u}}{\partial t} = -\frac{\partial h}{\partial x} - c_{21} \frac{\partial^3 h}{\partial x^3} + c_4 \frac{\partial^2 \bar{u}}{\partial x^2}. \quad (17)$$

Because of the diffusive dissipation, this system is not in the form (9) addressed by Theorem 1, yet we here demonstrate similar high order consistency between a gap-tooth scheme for this system and the PDEs.

The computer algebra program of Appendix A (with `choice:=2`) solves the PDEs (17) on patches coupled by (7). The resultant microscale fields in each patch and the discrete evolution are analogous to that obtained in section 4.2.1. Thus the equivalent PDE of the macroscale evolution is also the same as (16) except for additional terms introduced by the diffusive dissipation:

$$\frac{\partial H}{\partial t} = (\text{first three lines of (16)})$$

$$\begin{aligned}
& + c_3 \left[ \gamma^2 \frac{\partial^2 H}{\partial X^2} + \frac{1}{3}(\gamma^2 - \gamma^4) D^2 \frac{\partial^4 H}{\partial X^4} + \frac{2}{45}(\gamma^2 - 5\gamma^4 + 4\gamma^6) D^4 \frac{\partial^6 H}{\partial X^6} \right. \\
& \quad \left. + \frac{1}{315}(\gamma^2 - 21\gamma^4 + 56\gamma^6 - 36\gamma^8) D^6 \frac{\partial^8 H}{\partial X^8} \right] \\
& + \mathcal{O}((D^8 + c_{11} D^6) \frac{\partial^9 U}{\partial X^9} + c_3 D^8 \frac{\partial^{10} H}{\partial X^{10}}), \tag{18}
\end{aligned}$$

and similarly for  $\partial U / \partial t$ . As before, when the coupling between patches is limited to  $p - 1$  nearest neighbouring patches on either side, equivalent to truncating the coupling (7) to errors  $\mathcal{O}(\gamma^p)$ , then the PDE (18) confirms that consistency holds to errors  $\mathcal{O}(\partial^p / \partial X^p)$  for  $p \in \{3, 5, 7, 9\}$ ; that is, to errors  $\mathcal{O}(\delta^p)$ . This establishes the consistency of the gap-tooth scheme for the class of dispersive wave systems (17) with diffusive dissipation.

#### 4.2.3 Nonlinear wave systems

We generally want to use the gap-tooth scheme for macroscale simulation of nonlinear microscale dynamics. This section explores the basic example nonlinear wave system

$$\frac{\partial h}{\partial t} = -\frac{\partial \bar{u}}{\partial x} \quad \text{and} \quad \frac{\partial \bar{u}}{\partial t} = -\frac{\partial h}{\partial x} - c_5 \bar{u} \frac{\partial \bar{u}}{\partial x}, \tag{19}$$

where  $\bar{u} \partial \bar{u} / \partial x$  in the second PDE is typical of the self-advection of momentum. Because of the nonlinearity, this system is not in the form (9) addressed by Theorem 1, yet again we here demonstrate consistency between a gap-tooth scheme for this system and the PDEs.

The computer algebra program of Appendix A (with `choice:=3`) solves the nonlinear PDEs (19) on patches coupled by (7). But the nonlinearity needs to be small, and the algebraic complexity is great, so the computer algebra here solves the PDEs (19) to errors  $\mathcal{O}(\gamma^7, c_5^3)$ . Expressions for the microscale field and evolution of the macroscale quantities are rather complicated, and so omitted. Instead, we just record part of the equivalent PDEs for the macroscale variables in order to indicate how consistency with the microscale develops. Limiting the expressions to errors  $\mathcal{O}(D^4)$ ,

$$\begin{aligned}
H_t = & -\gamma U_x + \frac{1}{6} D^2 \gamma (1 - \gamma) \left[ -(1 + \gamma) U_{xxx} \right. \\
& + c_5 \gamma^2 r^2 (-2 H_{xx} U_x + \gamma \{H_x U_{xx} + H_{xxx} U\}) + c_5^2 \gamma^3 r^2 (-2 U_x^3 \\
& - 2 U_x U_{xx} + \gamma \{5 U_{xx} U_x + 2 U^2 U_{xxx} + H_x^2 U_x + 3 H_{xx} H_x U\}) \left. \right] \\
& + \mathcal{O}(\gamma^7, c_5^3, D^4), \tag{20}
\end{aligned}$$

$$\begin{aligned}
U_t = & -\gamma H_x - \gamma^2 c_5 U U_x + \frac{1}{6} D^2 \gamma (1 - \gamma) \left[ (1 + \gamma) H_{xxx} \right. \\
& + c_5^2 \gamma^2 r^2 (+2 \gamma H_{xx} U U_x + \{2 + 2\gamma - \gamma^2\} \{H_x U U_{xx} + H_{xxx} U^2\}) \left. \right] \\
& - c_5 D^2 \gamma^2 \frac{1}{24} (16 - 16\gamma^2 + \gamma^4 \{9 - 10r^2 + r^4\}) U U_{xxx} \\
& + \mathcal{O}(\gamma^7, c_5^3, D^4). \tag{21}
\end{aligned}$$

These equivalent PDEs for the macroscale dynamics of the gap-tooth scheme have several interesting aspects. The components  $H_t = -\gamma U_x$  and  $U_t = -\gamma H_x - \gamma^2 c_5 U U_x$  show that provided the inter-patch coupling involves at least the next nearest neighbours (flagged by the  $\gamma^2$ -factor), then the macroscale dynamics of the gap-tooth scheme is consistent with the microscale (19) to errors indicated by the other terms. The error terms of the  $H_t$ -PDE (20) all have a factor  $(1 - \gamma)$  so that

these errors vanish when the inter-patch coupling is carried out to high enough order: consequently consistency follows to  $\mathcal{O}(D^4)$ . Similarly for the first group of error terms of the  $\mathcal{U}_t$ -PDE (21). However, the second group of error terms of (21), that in  $c_5 D^2$ , do not appear to have a factor  $(1 - \gamma)$ , and so need not vanish: potentially, higher order analysis could find the requisite factor and remove the error; alternatively, the error vanishes for the case of cubic interpolation when truncating inter-patch coupling (7) to  $\mathcal{O}(\gamma^5)$ . Interestingly, this consistency error also vanishes for the overlapping patch case  $r = 1$  that is so attractive for ‘holistic discretisation’ (Roberts et al. 2013, e.g.), but overlapping patches are not relevant for efficient numerical simulation using the gap-tooth scheme. Summarising simply, the macroscale dynamics of the gap-tooth scheme is consistent with the microscale nonlinear dynamics of (19) to errors  $\mathcal{O}(D^2)$ , at most.

### 4.3 Nonlinear slow manifolds exist

The computer algebra of section 4.2 constructs the macroscale dynamics as a slow manifold of the gap-tooth scheme (Boyd 1995, Mackay 2004, e.g.). In the case of linear systems the slow manifolds are more specifically slow subspaces. This section establishes that such slow manifolds exist for some system close to that specified (Roberts 2014, Chapt. 13), and identifies that the fast microscale waves, if undamped, may nonetheless affect the macroscale dynamics.

We establish a gap-tooth slow manifold for microscale systems in the general form

$$\partial_t \mathbf{h} = \mathcal{L}_1 \bar{\mathbf{u}} + \epsilon \mathbf{f}_1(\mathbf{h}, \bar{\mathbf{u}}) \quad \text{and} \quad \partial_t \bar{\mathbf{u}} = \mathcal{L}_2 \mathbf{h} + \epsilon \mathbf{f}_2(\mathbf{h}, \bar{\mathbf{u}}), \quad (22)$$

for fields  $\mathbf{h}(\mathbf{x}, t), \bar{\mathbf{u}}(\mathbf{x}, t) \in \mathbb{H}^2(\Omega)$ , some sufficiently smooth functions  $\mathbf{f}_\ell$ , and homogeneous operators  $\mathcal{L}_\ell$  satisfying the following fast-slow dichotomy: forming the operators  $\mathcal{K}_\ell$ , whether differential as in (2) or discrete as in (3), into the two combined operators

$$\mathcal{K}_\ell = \begin{bmatrix} 0 & \mathcal{L}_\ell \\ \mathcal{L}_{3-\ell} & 0 \end{bmatrix}$$

such that the second component satisfies Dirichlet boundary conditions at edges of some interval, then both  $\mathcal{K}_1, \mathcal{K}_2$  must have a zero eigenvalue with corresponding eigenvector  $(1, 0)$ , and all other eigenvalues  $\lambda$  must be bounded away from zero, in modulus,  $|\lambda| > \beta > 0$ . Form a gap-tooth system by letting  $\mathbf{h}_j(\mathbf{x}, t), \bar{\mathbf{u}}_j(\mathbf{x}, t) \in \mathbb{H}^2(E_j)$  denote the subgrid fields on the  $j$ th patch satisfying (22) in patches  $E_j$  with the coupling conditions (8). Because of the zero eigenvalue of  $\mathcal{K}_\ell$ , for no nonlinearity,  $\epsilon = 0$ , and no coupling,  $\gamma = 0$ , this gap-tooth system has a subspace of equilibria  $\mathbb{M}_0$  of piecewise constant fields in each patch:

$$(\mathbf{h}_j, \bar{\mathbf{u}}_j) = (\mathbf{H}_j, 0) \text{ for odd } j, \quad (\mathbf{h}_j, \bar{\mathbf{u}}_j) = (0, \mathbf{U}_j) \text{ for even } j,$$

for independent ‘amplitudes’  $\mathbf{H}_j$  and  $\mathbf{U}_j$ . Set the eigenvalue bound  $\beta$  to be the smallest necessary for the range of  $\mathbb{M}_0$  of interest; typically  $\beta \propto 1/r$ . Then a power series construction finds the following (Cox & Roberts 1995, §3), (Roberts 2014, Chapt. 13): based at each of these equilibria there exists a smooth system and a smooth coordinate transformation which, firstly, together are  $\mathcal{O}(\gamma^p, \epsilon^q)$ -close to the gap-tooth system (22) with coupling (8), and secondly, possesses a slow manifold  $\mathcal{M}$  global in  $(\mathbf{H}, \mathbf{U})$ .<sup>2</sup> Section 4.2 constructed and discussed that

<sup>2</sup>Notice that this statement is a ‘backwards theory’ (Grcar 2011, e.g.) that neatly sidesteps the controversy over the existence or otherwise of slow manifolds (Lorenz 1986, Lorenz & Krishnamurty 1987, Lorenz 1992, Jacobs 1991, e.g.).



part of the coordinate transform that was on the slow manifold itself. This theory asserts that the macroscale grid values  $H_j$  and  $U_j$  discussed in section 4.2 can form a sound closure for a finite range of nonlinearity  $\epsilon$  and coupling  $\gamma$ .

However, for wave systems there is no assurance that the slow manifold is emergent. For nonlinear wave-like systems, the long term evolution on and off the slow manifold  $\mathcal{M}$  may be different—generally different by an amount quadratic in the fast waves (Cox & Roberts 1995). A user needs to be wary if the fast microscale waves in the patches  $E$  persist as a significant feature of the dynamics. The possibility is that then such fast waves ‘trapped’ in  $E$ , through resonance, may affect the macroscale evolution in a way significantly different from the way such fast waves affect the evolution if distributed over all space  $\Omega$ . In our simulations of turbulent floods this is not a problem as the turbulent dissipation,  $h|\bar{u}|_{xx}$ , that is so weak as to be typically negligible over interesting macroscales, is reasonably strong inside the microscale patches and damps the microscale fast waves. Similarly, in many applications some dissipation that is negligible on the macroscale will be a significant dissipation on the microscale and so damps the fast waves to leave the system on the slow manifold  $\mathcal{M}$ .

## 5 Gap-tooth simulation of dam breaking

This section applies the gap-tooth scheme to simulate dam-breaking. The aim is to show how the scheme caters for the extreme nonlinearity of the dam and turbulent bore discontinuities (section 5.1) and to discuss ways to implement practical domain boundary conditions (section 5.2) rather than the periodic domain used in previous Sections 3 and 4.

Figure 7 shows a dam that holds back water standing in the middle of the domain of length  $L$ . The ground is horizontal and let  $x$  denote the horizontal position. Initially the dam holds water upstream of nondimensional depth  $h = 1$ . To avoid poor conditioning in the numerical calculation, downstream of the dam let the water have a shallow depth, for example 0.1. At time  $t = 0$  the dam breaks and the upstream water rushes downstream.

We simulate the dam-breaking waves by the gap-tooth scheme with the microscale turbulent model (2) and the cubic coupling conditions (5). For comparison with the gap-tooth simulations, we also compute the microscale simulation over the whole domain, and report experimental data from Stansby et al. (1998) This section shows that the gap-tooth scheme reasonably simulates the dam-breaking waves.

There are typically two ways to distribute patches in the macroscale domain: either a patch includes the dam and the microscale resolves the sharp change in water depth, or the dam lies between two patches and the depth change is resolved only in the macroscale interpolation. Case 1 of Figure 7 distributes six patches in the macroscale domain. The dam stands in the middle of the gap between the third and fourth patches. An advantage of such a choice is to avoid the sharp discontinuity at the dam, because it arises in the gap which is not represented in the gap-tooth simulation. Case 2 in Figure 7 distributes five patches in the whole domain. The dam is included on the centre of the third patch. This choice would resolve the microscale details of the dynamics at the dam when the dam breaks.



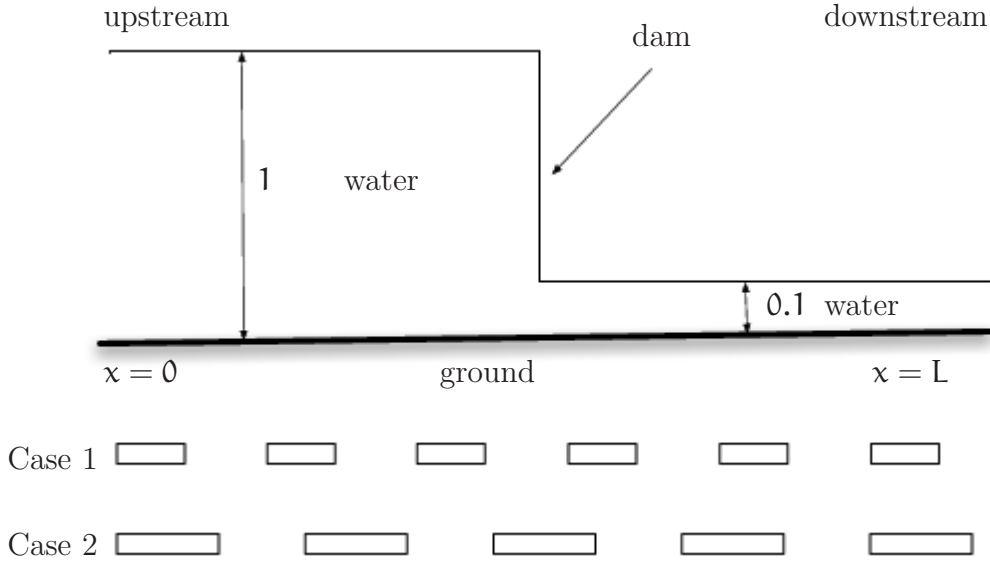


Figure 7: The initial conditions of the dam breaking on a domain of nondimensional length  $L$ . The dam located at  $x = L/2$  holds water of nondimensional depth  $h = 1$  in the upstream and a nondimensional shallow depth  $h = 0.1$  in the downstream. Case 1 places the dam lies in between patches, and Case 2 places the dam in the middle of a patch.

## 5.1 Numerical gap-tooth simulation of dam breaking

This section explores numerical gap-tooth simulations of the dam-breaking waves. Georgiev et al. (2009) used a previous version of the turbulent water wave PDEs (2) in simulating dam breaking; but they simulated the system over all space in the domain, not by the gap-tooth scheme. The comparison of the calculations and experimental data by Georgiev et al. (2009) shows that the PDEs (2) are a reasonable model of dam breaking waves. This section compares the gap-tooth simulation with the microscale simulation over the whole space domain, and with some experimental data of Stansby et al. (1998). The simulations indicate that putting the dam within a patch appears better.

We compute both the gap-tooth simulation and the microscale simulation over the whole domain for the dam-breaking waves. In the experiments by Stansby et al. (1998), the initial water depth behind the dam is 10 cm and lies in a horizontal domain of length of 200 cm. The dam stands at the centre of the domain. For comparison, we nondimensionalise the depth 10 cm to one; then the nondimensional length is  $L = 200 \text{ cm} / 10 \text{ cm} = 20$ . In this pilot study of gap-tooth simulation, we distributed both  $m = 10$  and  $m = 22$  patches on the whole macroscale domain and use  $n = 9$  microscale grid points on a patch:  $m = 10$  patches is low resolution of the macroscale, and  $m = 22$  patches is only a medium resolution. Then the distance between neighbouring patches is  $D = L/m = 2$ . For the low resolution simulation, using the scale ratio  $r = 1/6$ , the width of each patch is  $\ell = 2rD = 0.67$ , and the microscale grid step in each patch is  $d = \ell/(n + 1) \approx 0.07$ . For consistent comparison, let the spatial step in the microscale simulation over the whole macroscale space domain (not by the gap-tooth scheme) have the same microscale spatial step  $d$ .

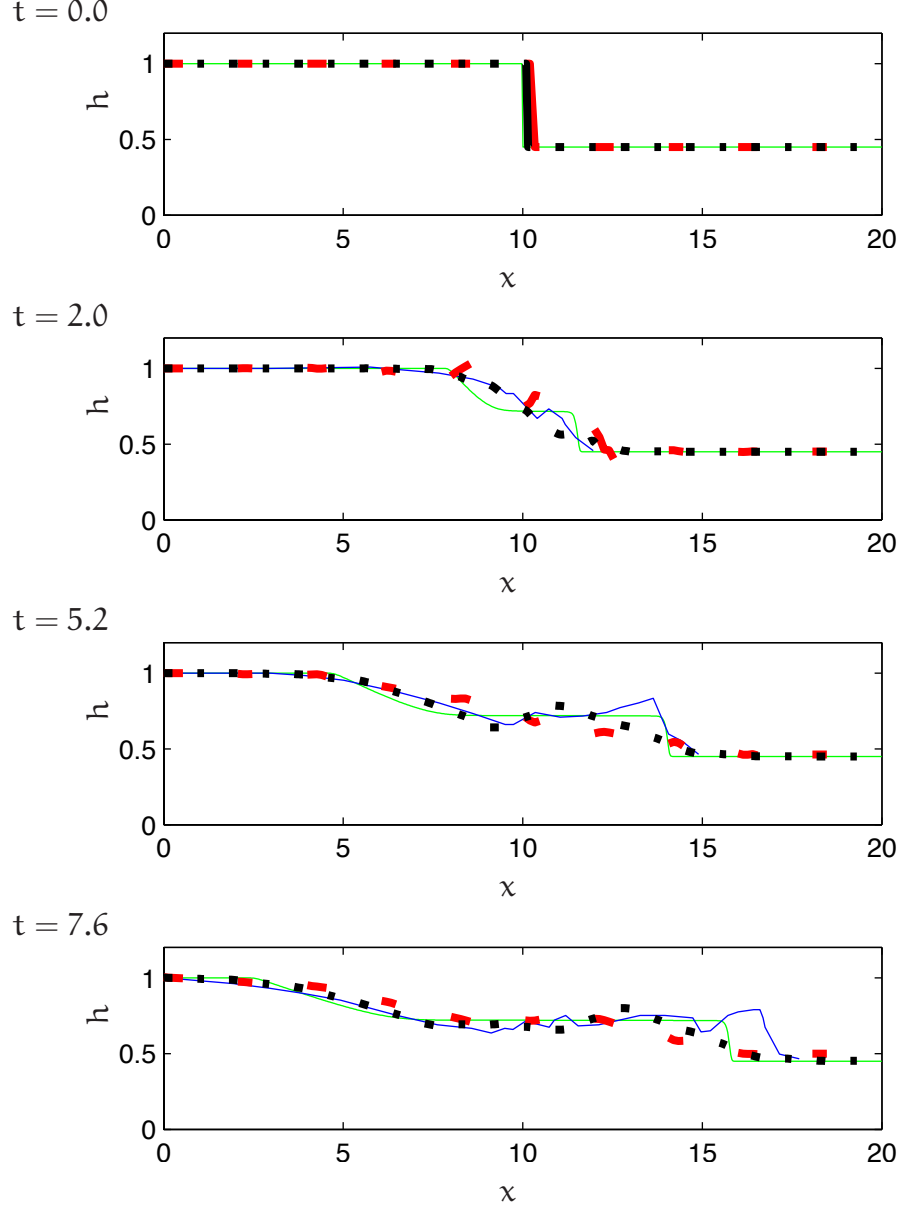


Figure 8: Comparison among the simulations of dam breaking: (red) low resolution gap-tooth with  $m = 10$  patches; (black) medium resolution gap-tooth with  $m = 22$  patches; (green) the microscale simulation over the whole domain; and (blue) some experimental data (Stansby et al. 1998, Fig. 8c). The dam is inside a patch. The nondimensional shallow depth is  $h = 0.45$  in front of the dam. The scale ratio  $r = 1/6$ , and the microscale step  $d = 1/15$ .

**The case of the dam being within a patch** Figure 8 plots at four times the gap-tooth simulation (red for  $m = 10$  patches and black for  $m = 22$  patches), the microscale simulation over the whole domain (green), and the experimental data (blue). The time  $t = 0$  graph shows that the water has depth one upstream and depth 0.45 downstream, which corresponds to the depth ratio of 0.45 in the experiments (Stansby et al. 1998, Fig.8c). Recall the nondimensional length scale  $H$ , velocity  $\sqrt{gH}$ , and time  $\sqrt{H/g}$ . Nondimensionalise the time in the experiments by  $\sqrt{0.1 \text{ m} / (10 \text{ m/s}^2)} = 0.1 \text{ s}$ . Therefore the plots of Stansby et al. (1998) are at nondimensional times  $t = 0, 2, 5.2$  and  $7.6$ .

The  $t = 2, 5.2, 7.6$  graphs in Figure 8 show that a turbulent bore forms in all simulations. However, the turbulent bore in the gap-tooth simulation is smoothed by the relatively large spacing between patches: the medium resolution simulation being noticeably more better than the low resolution. The bore in the gap-tooth simulation lags that in the experiment, while the bore in the microscale simulation over the whole domain reasonably tracks that in the experiment until the last time. The gap-tooth simulation has error  $\mathcal{O}(D^2) \sim 0.8$  for the medium resolution macroscale step  $D = L/m = 20/22 \approx 0.9$ , while the microscale simulation over the whole macroscale domain involves the error  $\mathcal{O}(d^2) \sim 0.001$  for the microscale step  $d = 2rD/(n+1) = 0.03$ . When the number of patches increases, the error  $\mathcal{O}(D^2)$  decreases, then the gap-tooth simulation performs better, as shown by the black data in comparison to the coarser red data.

However, the gap-tooth simulation saves computer time. The gap-tooth scheme only takes a compute time of 0.78 s for  $m = 22$  patches to simulate to the  $t = 7.6$  graph in Figure 8 (all simulations used Matlab with `ode15s` for time integration). Whereas the microscale simulation over the whole domain with the same microscale step  $d = 0.03$  needed a compute time of 74.3 s to simulate over the same time. That is, the whole domain simulation is nearly a hundred times slower than the gap-tooth simulation. Such a speed-up in this simple pilot study suggests, especially with smaller ratio  $r$  and implementing projective integration Gear & Kevrekidis (2003), Kevrekidis & Samaey (2009), that the gap-tooth scheme may empower simulation and analysis of large scale problems that are otherwise inaccessible.

Figure 9 shows in time the water area of the graphs in Figure 8. Since water is conserved, ideally these curves should be horizontal as seen for the green curve of the microscale simulation over the whole domain. The red curve shows that the low resolution ( $m = 10$  patches) gap-tooth simulation loses about 6% area, mainly in the initial dam break. The medium resolution simulation loses 2% (black curve), and again mostly in the initial dam break. Both of these losses are due to the relatively coarse spacing of the patches. The blue curve shows that the experiments gained fluid area, possibly due to entrainment of air in the turbulent bore (Stansby et al. 1998, Fig. 8c).

A further experiment had significantly shallower water in front of the dam which we also simulated. To avoid singularities being generated in the simulations near the dam, the dam was smoothed to a hyperbolic tangent and the smaller patch ratio  $r = 1/8$  was used. Figure 10 plots the gap-tooth simulation (red and black curves for  $m = 10$  and  $m = 22$  patches respectively), the microscale simulation over the whole macroscale domain (green curve), and the experimental data (blue curve) (Stansby et al. 1998, Fig. 8b). The initial shallow depth in front of the dam is 0.1 (nondimensional). Compared with Figure 8, the heights of the turbulent bore in the gap-tooth simulations are significantly smaller than that in the experiment and microscale simulation over the whole domain. Again, this seems due to the

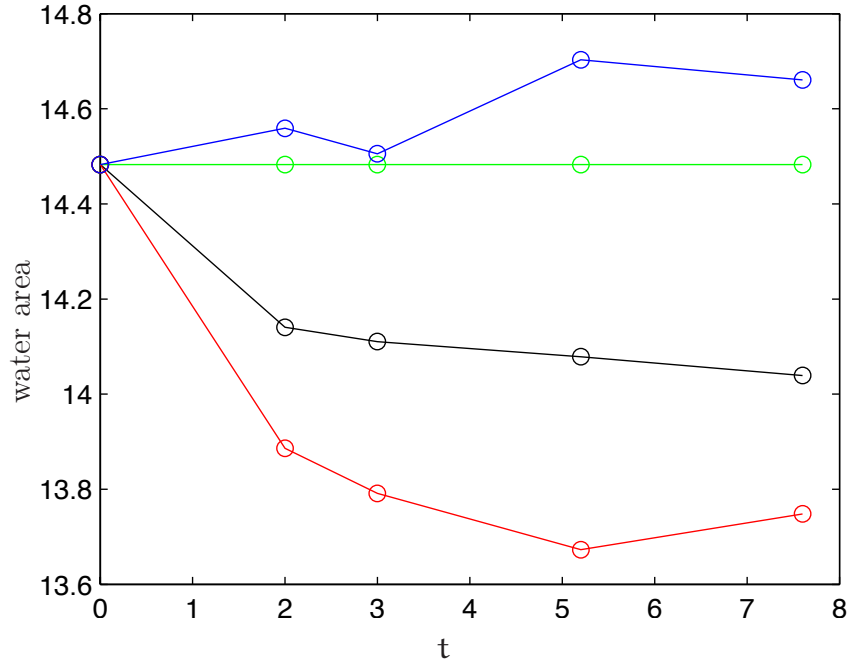


Figure 9: The water area over time of the simulations in Figure 8: (black) gap-tooth with  $m = 22$  patches; gap-tooth with (red)  $m = 10$  patches; (green) the microscale simulations over the whole space domain; and (blue) experiments by [Stansby et al. \(1998\)](#).

relatively coarse macroscale resolution in these gap-tooth simulations. The gap-tooth simulation should better approximate the dam-breaking waves with more patches or deeper water in front of the dam.

**Place the dam between two patches** Figure 11 plots the gap-tooth simulation (red and black curves for  $m = 10$  and  $m = 22$  patches respectively), the microscale simulation over the whole macroscale domain (green curve), and the experimental data (blue curve) by [Stansby et al. \(1998\)](#) at four times. The  $t = 0$  graph shows the initial depth, corresponding to the depth ratio of 0.45 in the experiments. The dam is not resolved within a patch in these gap-tooth simulation. The  $t = 2$  graph indicates that this gap-tooth simulation does not appear to be as good as the corresponding results in Figure 8. Then the  $t = 5.2$  graph shows there seems to be significantly more microscale oscillations in the low resolution case when compared to Figure 8. It appears that putting the dam within a patch is better.

## 5.2 Boundary conditions for gap-tooth simulation

This subsection discusses the invoked boundary conditions at the upstream and downstream boundaries of the macroscale domain in the gap-tooth simulation of the dam-breaking. Such boundary conditions will be needed in general simulations.

The gap-tooth simulation of the dam-breaking requires boundary conditions at the upstream  $x = 0$  and downstream  $x = L$ . Typically, no-flow boundary conditions are usually implemented in dam breaking ([Abdolmaleki et al. 2004](#), [Ozgokmen et al. 2007](#), e.g.). We consider three types of boundary condition at the upstream  $x = 0$  and downstream  $x = L$ :

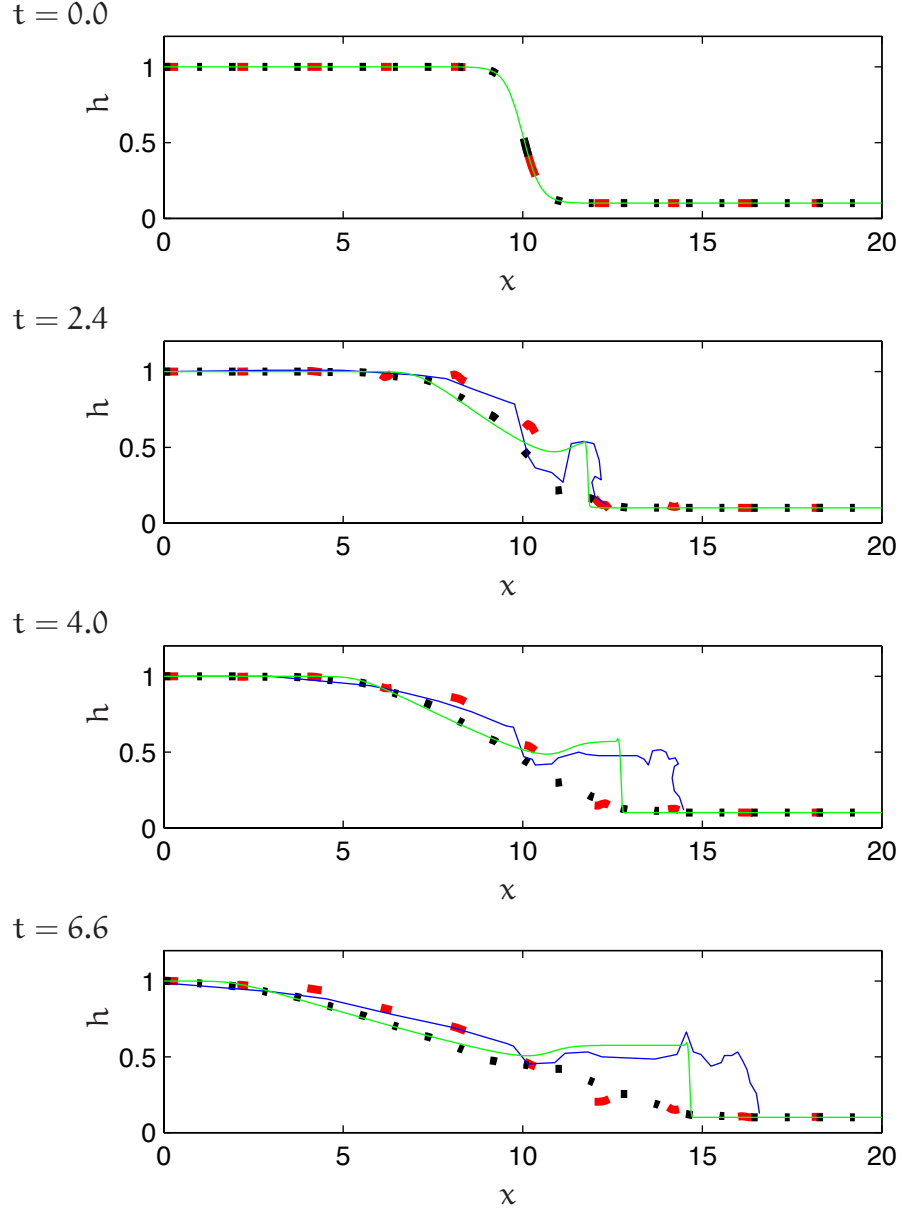


Figure 10: Comparison among the simulations of dam-breaking into shallow water of depth  $h = 0.1$ : (red) low resolution gap-tooth with  $m = 10$  patches; (black) medium resolution gap-tooth with  $m = 22$  patches; (green) the microscale simulation over the whole domain; and (blue) some experimental data (Stansby et al. 1998, Fig. 8c). The patch ratio  $r = 1/8$ , and the microscale step  $d = 1/20$ .

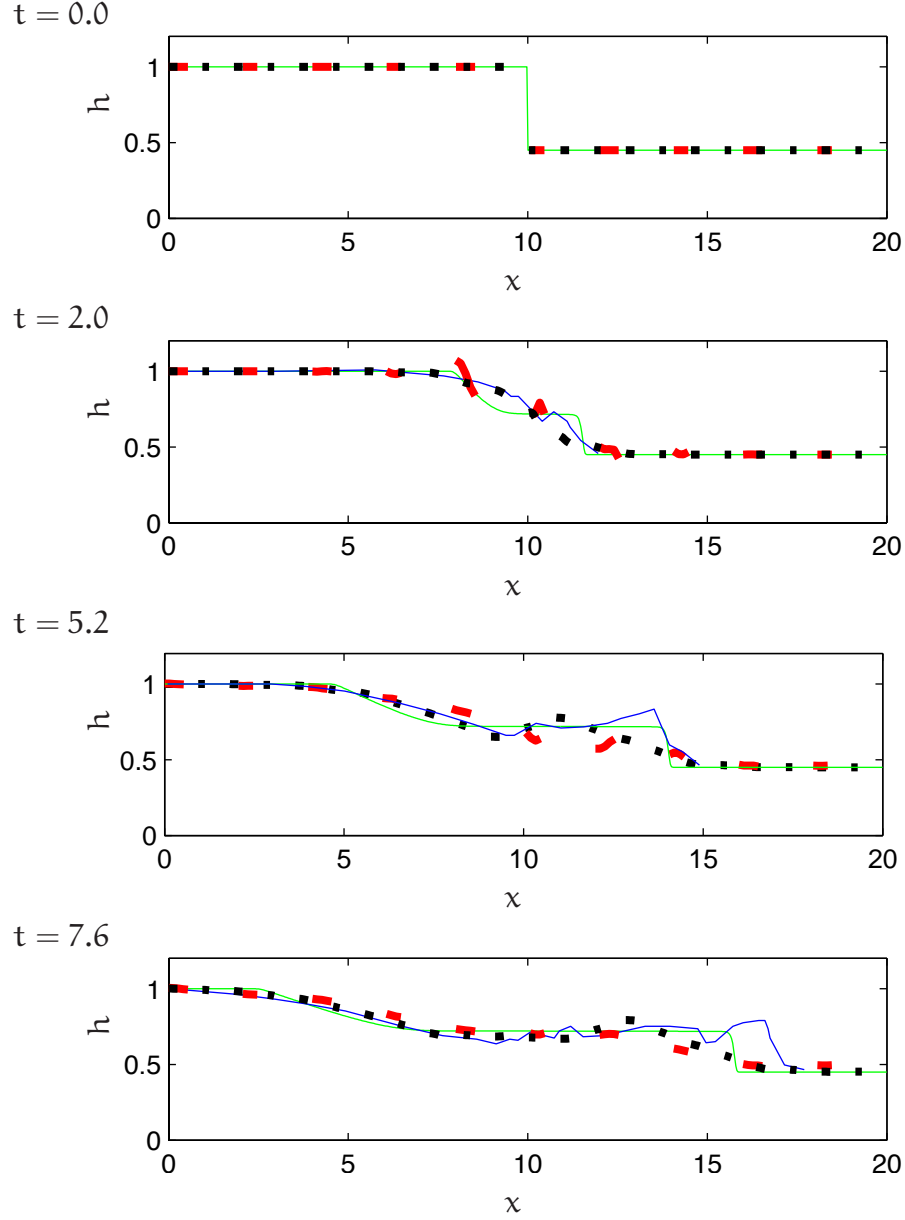


Figure 11: Comparison among the simulations of dam breaking: (red) low resolution gap-tooth with  $m = 10$  patches; (black) medium resolution gap-tooth with  $m = 22$  patches; (green) the microscale simulation over the whole domain; and (blue) the experimental data (Stansby et al. 1998, Fig. 8c). Initially the dam lies in between patches. The nondimensional shallow depth is  $h = 0.45$  in the front of the dam; the patch ratio  $r = 1/6$ ; and the microscale step  $d = 1/15$ .

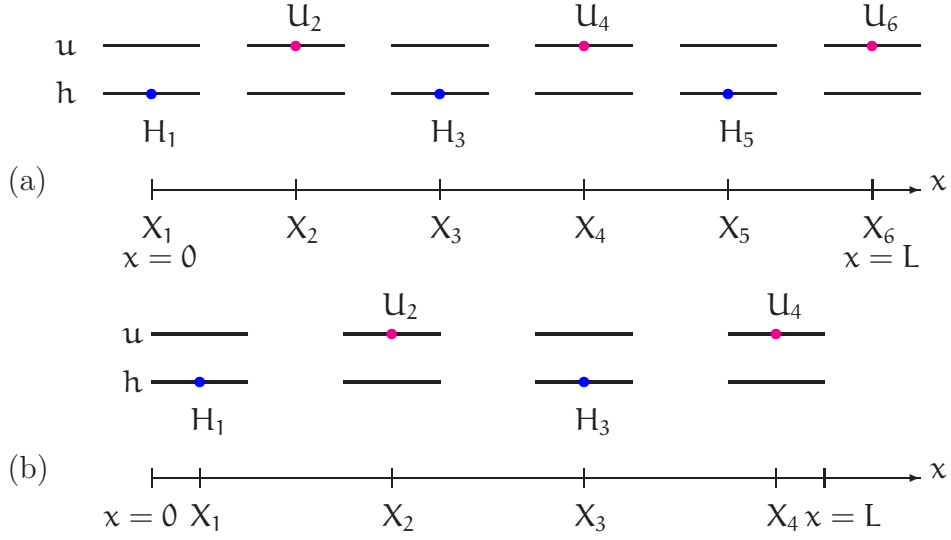


Figure 12: There are at least two ways to implement boundary conditions at the upstream  $x = 0$  and downstream  $x = L$  in gap-tooth simulation: (a)/(b), respectively, invokes the boundary conditions for the macroscale/microscale values of the leftmost and rightmost patches.

- constant depth such as  $h = 1$  at the upstream or depth  $h = 0.45$  at the downstream;
- no flux,  $\partial \bar{u} / \partial x = 0$ , at the upstream or at the downstream;
- and zero turbulent mean velocity,  $\bar{u} = 0$ , at the upstream or at the downstream, which is equivalent to no fluid flowing through the upstream or downstream,  $\partial h / \partial x = 0$  according to the momentum PDE (2b).

Since these boundary conditions could be applied either at the centre or edge of a patch, there could be at least  $\binom{3}{2}^2 = 36$  possible combinations of boundary conditions in the gap-tooth simulation.

In the gap-tooth simulation, boundary conditions are invoked to either the macroscale or microscale values on the leftmost and rightmost patches, as shown schematically by Figure 12. Figure 8–11 implement the boundary conditions  $\bar{u}_{1,1} = 0$  as drawn in Figure 12(b), and  $h_{m,1} = h_{m,n+2} = H_{m-1}$ , where  $\bar{u}_{1,1}$  is the left edge of the first patch,  $h_{m,1}$  and  $h_{m,n+2}$  are the edges of the  $m$ th patch, and  $H_{m-1}$  is the macroscale value on the  $(m-1)$ th patch through the coupling conditions (8) with the assumption of zero values on the fictitious  $(m+1)$ th patch in the simulation. Further work could explore the gap-tooth simulation with different boundary conditions.

## 6 Conclusion

Developing some preliminary research (Cao & Roberts 2013), we explored the gap-tooth scheme both theoretically and with a highly nonlinear microscale simulator (3) of turbulent shallow water waves (Section 2). The resultant numerical simulations indicated that the gap-tooth scheme on a staggered macroscale grid is useful for wave-like systems. Section 3 reported numerical eigenvalue analysis



that clearly showed, in Figure 5, the separation between relatively slow macroscale wave modes and the microscale fast waves supported within patches. The theoretical support of section 4.1 proves for a wide range of dispersive linear wave-like systems that the gap-tooth scheme generates macroscale simulations consistent with the microscale. Such consistency holds for a much wider class of nonlinear wave-like systems (section 4.2). Section 4.3 establishes that the gap-tooth scheme has a sound closure in terms of macroscale variables, but with the caveat that resonance among significant microscale waves could cause differing macroscale simulations. Section 5 applied the gap-tooth scheme to the highly nonlinear flow of dam-breaking waves. Figure 8–11 shows that although the turbulent bore lags and the height of this bore is a bit smaller, we reasonably predict the dam-breaking. The major limitation in the gap-tooth scheme appears to be that it cannot resolve microscale dynamics between the patches so it is primarily useful for macroscale dynamics which are globally varying on the macroscale. Nonetheless, there is scope for resolving microscale dynamics by, for example, putting the rapid changes associated with the dam break within a patch (section 5.1).

**Acknowledgements** Part of this research was supported by grant DP120104260 from the Australian Research Council. We thank Prof. Yannis Kevrekidis for inspiring discussions.

## References

- Abdolmaleki, K., Thiagarajan, K. P. & Morris-Thomas, M. T. (2004), Simulation of the dam break problem and impact flows using a Navier–Stokes solver. Proceedings of the Fifteenth Australasian Fluid Mechanics Conference (CD-ROM), editors M. Behnia, W. Lin, and G. D. McBain, The University of Sydney. <http://www.aeromech.usyd.edu.au/15afmc/proceedings>.
- Bijvelds, M. D. J. P., Kranenburg, C. & Stelling, G. S. (1999), ‘3D numerical simulation of turbulent shallow-water in square harbor’, *Journal of Hydraulic Engineering* **125**(1), 26–31. doi:10.1061/(ASCE)0733-9429(1999)125:1(26).
- Boyd, J. P. (1995), ‘Eight definitions of the slow manifold: seiches, pseudoseiches and exponential smallness’, *Dynamics of Atmospheres and Oceans* **22**(1–2), 49–75. <http://www.sciencedirect.com/science/article/B6VCR-3YCMH9H-B/2/1a3f0392189659c07>
- Cao, M. (2014), Modelling environmental turbulent fluids and multiscale modelling couples patches of wave-like system, PhD thesis, School of Mathematical Sciences, University of Adelaide.
- Cao, M. & Roberts, A. J. (2012), Modelling 3D turbulent floods based upon the Smagorinski large eddy closure. Proceedings of the 18th Australasian Fluid Mechanics Conference 3rd-7th December 2012 Edited by P. A. Brandner and B. W. Pearce Published by the Australasian Fluid Mechanics Society. <http://www.afms.org.au/proceedings/18AFMCTOC.htm>.
- Cao, M. & Roberts, A. J. (2013), Multiscale modelling couples patches of wave-like simulations, in S. McCue, T. Moroney, D. Mallet & J. Bunder,

- eds, ‘Proceedings of the 16th Biennial Computational Techniques and Applications Conference, CTAC-2012’, Vol. 54 of *ANZIAM J.*, pp. C153–C170. <http://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/6137> [May 14, 2013].
- Chicone, C. (2006), *Ordinary Differential Equations with Applications*, Vol. 34 of *Texts in Applied Mathematics*, Springer New York. [doi:10.1007/0-387-35794-7](https://doi.org/10.1007/0-387-35794-7).
- Cox, S. M. & Roberts, A. J. (1995), ‘Initial conditions for models of dynamical systems’, *Physica D* **85**, 126–141.
- E, W. & Engquist, B. (2003), ‘The heterogeneous multiscale methods’, *Comm. Math. Sci.* **1**(1), 87–132. <http://projecteuclid.org/euclid.cms/1118150402>.
- Fateman, R. (2003), ‘Comparing the speed of programs for sparse polynomial multiplication’, *ACM SIGSAM Bulletin* **37**(1), 4–15. <http://www.cs.berkeley.edu/~fateman/papers/fastmult.pdf>
- Gear, C. W. & Kevrekidis, I. G. (2003), ‘Projective methods for stiff differential equations: Problems with gaps in their eigenvalue spectrum’, *SIAM Journal on Scientific Computing* **24**(4), 1091–1106. <http://link.aip.org/link/?SCE/24/1091/1>
- Gear, C. W., Li, J. & Kevrekidis, I. G. (2003), ‘The gap-tooth method in particle simulations’, *Phys. Lett. A* **316**, 190–195. [doi:10.1016/j.physleta.2003.07.004](https://doi.org/10.1016/j.physleta.2003.07.004).
- Georgiev, D. J., Roberts, A. J. & Strunin, D. V. (2009), Modelling turbulent flow from dam break using slow manifolds, in G. N. Mercer & A. J. Roberts, eds, ‘Proceedings of the 14th Biennial Computational Techniques and Applications Conference, CTAC-2008’, Vol. 50 of *ANZIAM J.*, pp. C1033–C1051. <http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/1466> [September 3, 2009].
- Grcar, J. F. (2011), ‘John von Neumann’s analysis of Gaussian elimination and the origins of modern numerical analysis’, *SIAM Review* **53**(4), 607–682. <http://www.siam.org/journals/sirev/53-4/73471.html>
- Hou, T. Y., Yang, D. & Ran, H. (2008), ‘Multiscale analysis and computation for the three-dimensional incompressible Navier-Stokes equations’, *Multiscale Modelling and Simulation* **6**(4), 1317–1346. [doi:10.1137/070682046](https://doi.org/10.1137/070682046).
- Jacobs, S. J. (1991), ‘Existence of a slow manifold in a model system of equations’, *J. Atmos. Sci.* **48**, 893–901.
- Kevrekidis, I. G., Gear, G. W., Hyman, J. M., Kevrekidis, P. G., Runborg, O. & Theodoropoulos, C. (2003), ‘Equation-free, coarse-grained multiscale computation: enabling microscopic simulators to perform system-level analysis’, *Comm. Math. Sci.* **1**(4), 715–762. <http://projecteuclid.org/euclid.cms/1119655353>.

- Kevrekidis, I. G. & Samaey, G. (2009), ‘Equation-free multiscale computation: Algorithms and applications’, *Annu. Rev. Phys. Chem.* **60**, 321–44.  
<http://dx.doi.org/10.1146/annurev.physchem.59.032607.093610>
- LeVeque, R. J., George, D. L. & Berger, M. J. (2011), ‘Tsunami modelling with adaptively refined finite volume methods’, *Acta Numerica* **20**, 211–289.  
[doi:10.1017/S0962492911000043](https://doi.org/10.1017/S0962492911000043).
- Liu, H., Zhou, G. J. & Burrows, R. (2009), ‘Lattice boltzmann model for shallow water flows in curved and meandering channels’, *International Journal of Computational Fluid Dynamics* **23**(3), 209–220.  
<http://www.tandfonline.com/doi/abs/10.1080/10618560902754924>
- Lorenz, E. & Krishnamurty (1987), ‘On the non-existence of a slow manifold’, *J. Atmos. Sci.* **44**, 2940–2950.
- Lorenz, E. N. (1986), ‘On the existence of a slow manifold’, *J. Atmos. Sci.* **43**, 1547–1557.
- Lorenz, E. N. (1992), ‘The slow manifold—what is it?’, *Journal of the Atmospheric Sciences* **49**(24), 2449–2451.
- Mackay, R. S. (2004), Slow manifolds, in T. Dauxois, A. Litvak-Hinenzon, R. S. MacKay & A. Spanoudaki, eds, ‘Energy Localisation and Transfer’, World Sci.
- Monaghan, J. J. (1992), ‘Smoothed particle hydrodynamics’, *Annu Rev Astron Astrophys* **30**, 543–574.
- Natl Physical Lab (1961), *Modern Computing Methods*, Vol. 16 of *Notes on Applied Science*, 2nd edn, Her Majesty’s Stationery Office, London.
- Ozgokmen, T. M., Iliescu, T., Fisher, P. F., Srinivasan, A. & Duan, J. (2007), ‘Large eddy simulation of stratified mixing in two-dimensional dam-break problem in a rectangular enclosed domain’, *Ocean Modelling* **16**, 106–140.  
[doi:10.1016/j.ocemod.2006.08.006](https://doi.org/10.1016/j.ocemod.2006.08.006).
- Potzsche, C. & Rasmussen, M. (2006), ‘Taylor approximation of integral manifolds’, *Journal of Dynamics and Differential Equations* **18**(2), 427–460.  
[doi:10.1007/s10884-006-9011-8](https://doi.org/10.1007/s10884-006-9011-8).
- Roberts, A. J. (1988), ‘The application of centre-manifold theory to the evolution of systems which vary slowly in space’, *J. Austral. Math. Soc. Ser. B* **29**, 480–500.  
[doi:10.1017/S0334270000005968](https://doi.org/10.1017/S0334270000005968).
- Roberts, A. J. (2001), ‘Holistic finite differences ensure fidelity to Burger’s equation’, *Applied Numerical Modelling* **37**, 371–396.  
[doi:10.1016/S0168-9274\(00\)00053-2](https://doi.org/10.1016/S0168-9274(00)00053-2).
- Roberts, A. J. (2003), ‘A holistic finite difference approach models linear dynamics consistently’, *Mathematics of Computation* **72**, 247–262.  
<http://www.ams.org/mcom/2003-72-241/S0025-5718-02-01448-5>
- Roberts, A. J. (2008), Computer algebra describes flow of turbulent floods via the smagorinski large eddy closure, Technical report.  
<http://eprints.usq.edu.au/4008/>.

Roberts, A. J. (2014), *Model emergent dynamics in complex systems*, 1st edn, SIAM, Philadelphia.

<http://www.maths.adelaide.edu.au/anthony.roberts/modelling.php>

Roberts, A. J. & Kevrekidis, I. G. (2005), ‘Higher order accuracy in the gap-tooth scheme for large-scale dynamics using microscopic simulators’, *ANZIAM Journal* **46**, C637–C657.

<http://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/981/847>.

Roberts, A. J., MacKenzie, T. & Bunder, J. (2013), ‘Accurate macroscale modelling of spatial dynamics in multiple dimensions’, *J. Engineering Mathematics Online* **01-Oct**.

<http://arxiv.org/abs/1103.1187>

Roberts, A. J., MacKenzie, T. & Bunder, J. E. (2011), ‘Accurate macroscale modelling of spatial dynamics in multiple dimensions’, *arXiv preprint arXiv:1103.1187*. <http://arxiv.org/abs/1103.1187>.

Samaey, G., Kevrekidis, I. G. & Roose, D. (2005), ‘The gap-tooth scheme for homogenization problems’, *SIAM Multiscale Modeling and Simulation* **4**, 278–306. doi:10.1137/030602046.

Samaey, G., Roberts, A. J. & Kevrekidis, I. G. (2009), *Equation-free computation: an overview of patch dynamics*, Vol. 8 of Multiscale Methods, Oxford Scholarship Online Monographs. doi:10.1093/acprof:oso/9780199233854.003.0008.

Samaey, G., Roose, D. & Kevrekidis, I. G. (2005), ‘The gap-tooth scheme for homogenization problems’, *Multiscale Modelling and Simulation* **4**, 278–306. doi:10.1137/030602046.

Southern, J., Pitt-Francis, J., Whiteley, J., Stokeley, D., Kobashi, H., Nobes, R., Kadooka, Y. & Gavaghan, D. (2008), ‘Multi-scale computational modelling in biology and physiology’, *Progress in Biophysics and Molecular Biology* **96**(1–3), 60–89.

<http://www.sciencedirect.com/science/article/B6TBN-4PD4X9M-9/2/09e50ee57acbaa383>

Stansby, P. K., Chagini, A. & Barnes, T. C. D. (1998), ‘The initial stages of dam-break flow’, *Journal of Fluid Mechanics* **374**, 407–424. doi:10.1017/S0022112098009975.

## A Ancillary computer algebra program

This computer algebra code constructs the slow manifolds for the gap-tooth scheme discussed in Section 4.2. We use the Reduce computer algebra package<sup>3</sup> because it is freely available and because it is perhaps the fastest general purpose computer algebra system (Fateman 2003, e.g.).

- 1 Comment Computer algebra proves the consistency of the
- 2 gap-tooth scheme to high orders for wave-like systems where
- 3 patches are coupled with classic interpolation.
- 4 Meng Cao and A. J. Roberts, 9 April 2014.

---

<sup>3</sup><http://www.reduce-algebra.com/>

```

5
6 Variable 'choice' chooses the PDEs.
7 1 is for  $(h,u)_t = (h,u) - (u,h)_x - (u,h)_{xxx} +$ 
8 2 is for  $(h,u)_t = (h,u)_{xx} - (u,h)_x - (u,h)_{xxx} +$ 
9 otherwise a nonlinear case
10 ;
11 choice:=3;
12 if choice<3 then let gamma^9=>0
13     else let{gamma^7=>0, c5^3=>0};
14
15 on div; off allfac; on revpri; linelength 70$
16 factor dd,df;
17 factor c1,c2,c11,c21,c3,c4,c5;
18
19 % Define integrate operator to handle polynomials.
20 operator intx; linear intx;
21 let { intx(xi^~p,xi)=>xi^(p+1)/(p+1)
22     , intx(1,xi)=>xi
23     , intx(xi^~p,xi,-1,1)=>(1-(-1)^(p+1))/(p+1)
24     , intx(1,xi,-1,1)=>2
25     };
26
27 % Introduce the subgrid variable  $xi=(x-X_j)/r/D$ ,  $|xi|<1$ .
28 depend xi,x;
29 let df(xi,x)=>1/dd/r;
30
31 % Define evolving amplitudes.
32 operator hh; operator uu;
33 depend hh,t; depend uu,t;
34 let { df(hh(~k),t)=>sub(j=k,gh1)
35     , df(uu(~k),t)=>sub(j=k,gu2)
36     };
37
38 % initial approximation
39 hj1:=hh(j);
40 uj2:=uj1:=0;
41 uj2:=uu(j);
42 gh1:=gu2:=0;
43
44 for it:=1:29 do begin
45 % update for the -1-field.
46 % residuals of the PDEs.
47 if choice=1 then begin
48 resh1:=df(hj1,t)+df(uj1,x)+c11*df(uj1,x,3)-c1*hj1;
49 resul:=df(uj1,t)+df(hj1,x)+c21*df(hj1,x,3)-c2*uj1;
50 end else if choice=2 then begin
51 resh1:=df(hj1,t)+df(uj1,x)+c11*df(uj1,x,3)-c3*df(hj1,x,2);
52 resul:=df(uj1,t)+df(hj1,x)+c21*df(hj1,x,3)-c4*df(uj1,x,2);
53 end else begin
54 resh1:=df(hj1,t)+df(uj1,x);
55 resul:=df(uj1,t)+df(hj1,x)+c5*df(uj1,x)*uj1;
56 end;
57 % residual of the coupling conditions, errors 0(gamma^9);

```

```

58 resu1r:=sub(xi=+1,u1)
59   -gamma*(uu(j+1)+uu(j-1))/2-gamma*r/2*(uu(j+1)-uu(j-1))
60   -gamma^3*(-1+r^2)/16*(uu(j+3)-uu(j+1)-uu(j-1)+uu(j-3))
61   -gamma^3*(-r+r^3)/48*(uu(j+3)-3*uu(j+1)+3*uu(j-1)-uu(j-3))
62   -gamma^5*(9-10*r^2+r^4)/768*(uu(j+5)+uu(j+3)-2*uu(j+1)
63     -2*uu(j-1)+uu(j-3)+uu(j-5))
64   -gamma^5*(9*r-10*r^3+r^5)/3840*(uu(j+5)-5*uu(j+3)+10*uu(j+1)
65     -10*uu(j-1)+5*uu(j-3)-uu(j-5))
66   -gamma^7*(-225+259*r^2-35*r^4+r^6)/92160*(uu(j+7)-5*uu(j+5)
67     +9*uu(j+3)-5*uu(j+1)-5*uu(j-1)+9*uu(j-3)-5*uu(j-5)+uu(j-7))
68   -gamma^7*(-225*r+259*r^3-35*r^5+r^7)/645120*(uu(j+7)
69     -7*uu(j+5)+21*uu(j+3)-35*uu(j+1)+35*uu(j-1)-21*uu(j-3)+7*uu(j-5)-uu(j-7));
70 resu1l:=sub(xi=-1,u1)
71   -gamma*(uu(j+1)+uu(j-1))/2+gamma*r/2*(uu(j+1)-uu(j-1))
72   -gamma^3*(-1+r^2)/16*(uu(j+3)-uu(j+1)-uu(j-1)+uu(j-3))
73   +gamma^3*(-r+r^3)/48*(uu(j+3)-3*uu(j+1)+3*uu(j-1)-uu(j-3))
74   -gamma^5*(9-10*r^2+r^4)/768*(uu(j+5)+uu(j+3)-2*uu(j+1)
75     -2*uu(j-1)+uu(j-3)+uu(j-5))
76   +gamma^5*(9*r-10*r^3+r^5)/3840*(uu(j+5)-5*uu(j+3)+10*uu(j+1)
77     -10*uu(j-1)+5*uu(j-3)-uu(j-5))
78   -gamma^7*(-225+259*r^2-35*r^4+r^6)/92160*(uu(j+7)-5*uu(j+5)
79     +9*uu(j+3)-5*uu(j+1)-5*uu(j-1)+9*uu(j-3)-5*uu(j-5)+uu(j-7))
80   +gamma^7*(-225*r+259*r^3-35*r^5+r^7)/645120*(uu(j+7)
81     -7*uu(j+5)+21*uu(j+3)-35*uu(j+1)+35*uu(j-1)-21*uu(j-3)+7*uu(j-5)-uu(j-7));
82 write lengthres1:=map(length(~a),{resh1,resu1,resu1r,resu1l});
83 % update gh1,u1 fields.
84 gh1:=gh1+(ghd1:=(resu1r-resu1l)/2/dd/r
85   -intx(resh1,xi,-1,1)/2);
86 hj1:=hj1-r*dd*intx(resu1,xi);
87 uj1:=uj1-r*dd*intx(resh1+ghd1,xi)-(resu1r+resu1l)/2;
88
89 % update for the -2-fields.
90 if choice=1 then begin
91   resh2:=df(hj2,t)+df(uj2,x)+c11*df(uj2,x,3)-c1*hj2;
92   resu2:=df(uj2,t)+df(hj2,x)+c21*df(hj2,x,3)-c2*uj2;
93 end else if choice=2 then begin
94   resh2:=df(hj2,t)+df(uj2,x)+c11*df(uj2,x,3)-c3*df(hj2,x,2);
95   resu2:=df(uj2,t)+df(hj2,x)+c21*df(hj2,x,3)-c4*df(uj2,x,2);
96 end else begin
97   resh2:=df(hj2,t)+df(uj2,x);
98   resu2:=df(uj2,t)+df(hj2,x)+c5*df(uj2,x)*uj2;
99 end;
100 % residual of the coupling conditions, errors 0(gamma^9);
101 resh2r:=sub(xi=+1,hj2)
102   -gamma*(hh(j+1)+hh(j-1))/2-gamma*r/2*(hh(j+1)-hh(j-1))
103   -gamma^3*(-1+r^2)/16*(hh(j+3)-hh(j+1)-hh(j-1)+hh(j-3))
104   -gamma^3*(-r+r^3)/48*(hh(j+3)-3*hh(j+1)+3*hh(j-1)-hh(j-3))
105   -gamma^5*(9-10*r^2+r^4)/768*(hh(j+5)+hh(j+3)-2*hh(j+1)
106     -2*hh(j-1)+hh(j-3)+hh(j-5))
107   -gamma^5*(9*r-10*r^3+r^5)/3840*(hh(j+5)-5*hh(j+3)+10*hh(j+1)
108     -10*hh(j-1)+5*hh(j-3)-hh(j-5))
109   -gamma^7*(-225+259*r^2-35*r^4+r^6)/92160*(hh(j+7)-5*hh(j+5)
110     +9*hh(j+3)-5*hh(j+1)-5*hh(j-1)+9*hh(j-3)-5*hh(j-5)+hh(j-7))

```



```

111 -gamma^7*(-225*r+259*r^3-35*r^5+r^7)/645120*(hh(j+7)
112 -7*hh(j+5)+21*hh(j+3)-35*hh(j+1)+35*hh(j-1)-21*hh(j-3)+7*hh(j-5)-hh(j-7));
113 resh2l:=sub(xi=-1,hj2)
114 -gamma*(hh(j+1)+hh(j-1))/2+gamma*r/2*(hh(j+1)-hh(j-1))
115 -gamma^3*(-1+r^2)/16*(hh(j+3)-hh(j+1)-hh(j-1)+hh(j-3))
116 +gamma^3*(-r+r^3)/48*(hh(j+3)-3*hh(j+1)+3*hh(j-1)-hh(j-3))
117 -gamma^5*(9-10*r^2+r^4)/768*(hh(j+5)+hh(j+3)-2*hh(j+1)
118 -2*hh(j-1)+hh(j-3)+hh(j-5))
119 +gamma^5*(9*r-10*r^3+r^5)/3840*(hh(j+5)-5*hh(j+3)+10*hh(j+1)
120 -10*hh(j-1)+5*hh(j-3)-hh(j-5))
121 -gamma^7*(-225+259*r^2-35*r^4+r^6)/92160*(hh(j+7)-5*hh(j+5)
122 +9*hh(j+3)-5*hh(j+1)-5*hh(j-1)+9*hh(j-3)-5*hh(j-5)+hh(j-7))
123 +gamma^7*(-225*r+259*r^3-35*r^5+r^7)/645120*(hh(j+7)
124 -7*hh(j+5)+21*hh(j+3)-35*hh(j+1)+35*hh(j-1)-21*hh(j-3)+7*hh(j-5)-hh(j-7));
125 write lengthres2:=map(length(~a),{resh2,resu2,resh2r,resh2l});
126 gu2:=gu2+(gud2:=(resh2r-resh2l)/2/dd/r
127 -intx(resu2,xi,-1,1)/2);
128 uj2:=uj2-r*dd*intx(resh2,xi);
129 hj2:=hj2-r*dd*intx(resu2+gud2,xi)-(resh2r+resh2l)/2;
130
131 showtime;
132 if {resh1,resu1,resh2,resu2,resu1l,resu1r,resh2l,resh2r}
133   ={0,0,0,0,0,0,0,0} then write it:=it+100000;
134 end;
135 if {resh1,resu1,resh2,resu2,resu1l,resu1r,resh2l,resh2r}
136   neq{0,0,0,0,0,0,0,0} then rederr "FAILED TO CONVERGE";
137
138 % Optionally convert soln to central difference operator form.
139 % This code not appropriate for nonlinear systems.
140 if choice<3 then begin
141   factor hh,uu;
142   rules:={ mu^2=>1+delta^2/4, uu(j)=>uu, hh(j)=>hh
143     , uu(j+~p)=>(mu+sign(p)*delta/2)^abs(p)*uu
144     , hh(j+~p)=>(mu+sign(p)*delta/2)^abs(p)*hh }$
145   write dhdt:=(gh1 where rules);
146   write dudt:=(gu2 where rules);
147   if 0 then begin factor delta,mu;
148     write h1op:=(hj1 where rules);
149     write u2op:=(uj2 where rules);
150     write h2op:=(hj2 where rules);
151     write u1op:=(uj1 where rules);
152   end;
153 end;
154
155 % Find the equivalent PDE to the discrete model
156 o:=deg((1+gamma)^20,gamma)+3; % order of truncation
157 operator h; operator u;
158 depend h,x; depend u,x;
159 factor h,u;
160 rules:={uu(j)=>u, uu(j+~p)=>u+(for n:=1:o sum
161     u(n)*(dd*p)^n/factorial(n))
162     ,hh(j)=>h, hh(j+~p)=>h+(for n:=1:o sum
163     h(n)*(dd*p)^n/factorial(n))

```



```

164         }$
165 ghde:=((gh1 where rules)where dd~p=>0 when p>=o);
166 gude:=((gu2 where rules)where dd~p=>0 when p>=o);
167 equivdes:=sub({gamma=1},{ghde,gude});
168 end;

```